Nonstandard Analysis: a new way to compute

Sam Sanders[1]

Model Theory and Proof Theory of Arithmetic

A Memorial Conference in Honour of H. Kotlarski and Z. Ratajczyk

July 25, 2012

UNIVERSITEIT GENT

TOHOKU UNIVERSITY

[1]This research is generously supported by the John Templeton Foundation.

Take-home message

Take-home message

In Nonstandard Analysis, an algorithm is any object whose definition is independent of the choice of infinitesimal (Ω-invariance).

Take-home message

In Nonstandard Analysis, an algorithm is any object whose definition is independent of the choice of infinitesimal (Ω-invariance).

More technically, we define a translation between Constructive Analysis (BISH) and Nonstandard Analysis (NSA):

Take-home message

In Nonstandard Analysis, an algorithm is any object whose definition is independent of the choice of infinitesimal (Ω-invariance).

More technically, we define a translation between Constructive Analysis (BISH) and Nonstandard Analysis (NSA):

(Proof and Algorithm) in BISH = (Transfer and Ω-invariance) in NSA

Take-home message

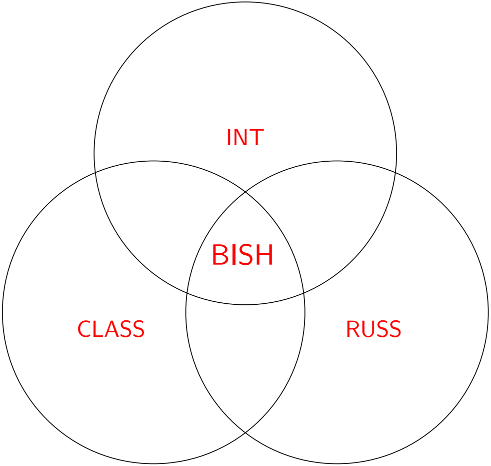
In Nonstandard Analysis, an algorithm is any object whose definition is independent of the choice of infinitesimal (Ω-invariance).

More technically, we define a translation between Constructive Analysis (BISH) and Nonstandard Analysis (NSA):

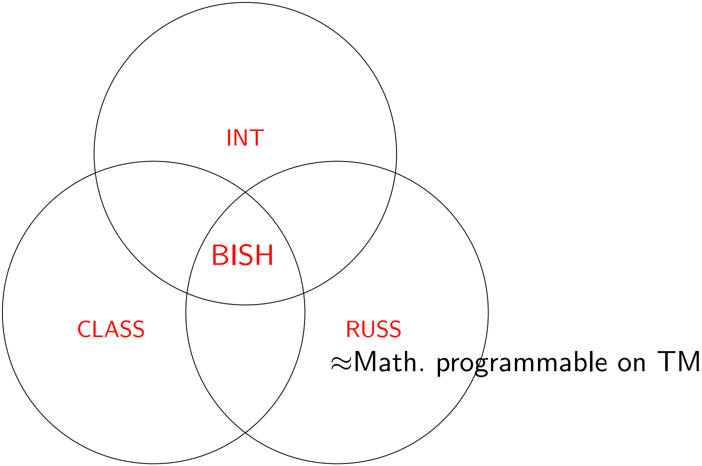
(Proof and Algorithm) in BISH = (Transfer and Ω-invariance) in NSA

Most results from CRM (= RM based on BISH) translate to NSA under a natural translation $\mathbb{B}$.

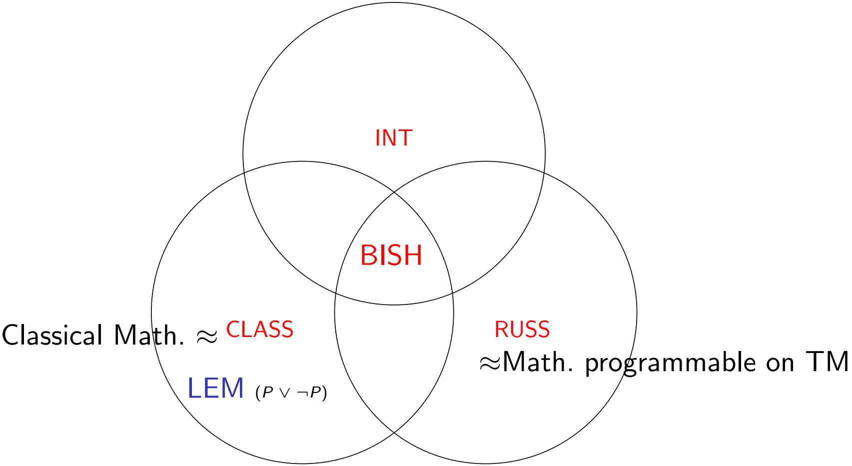
Son of a…



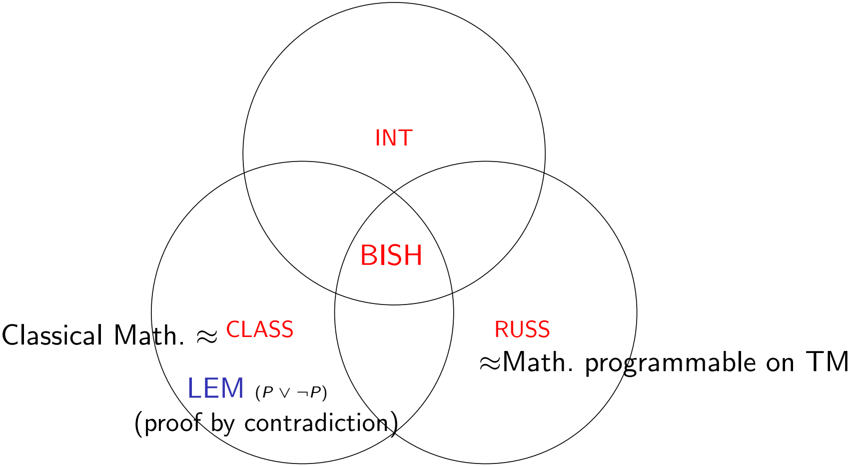
Son of a…



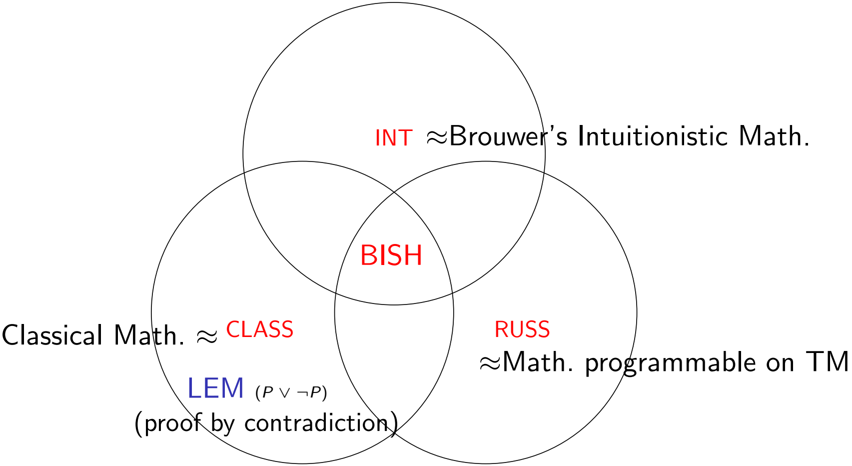
Son of a…


INT
BISH
Classical Math. ≈ CLASS
RUSS
≈Math. programmable on TM
LEM (P ∨ ¬P)

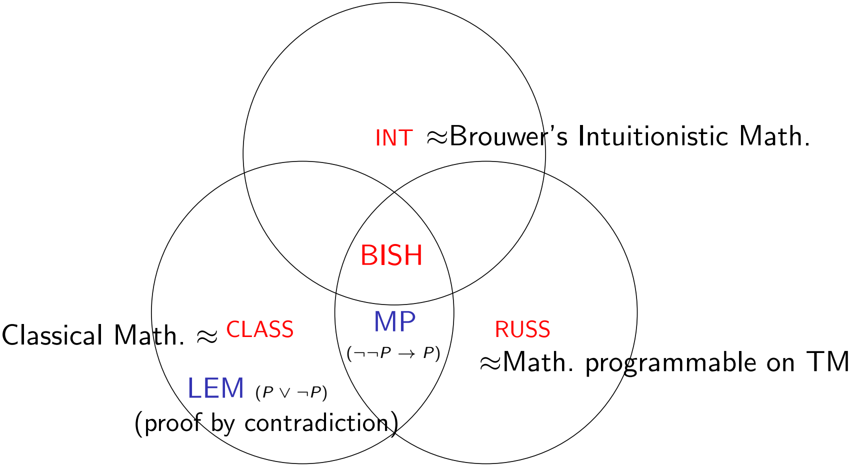
Son of a…


INT
BISH
CLASS
RUSS
Classical Math. ≈
≈Math. programmable on TM
LEM $(P \vee \neg P)$
(proof by contradiction)

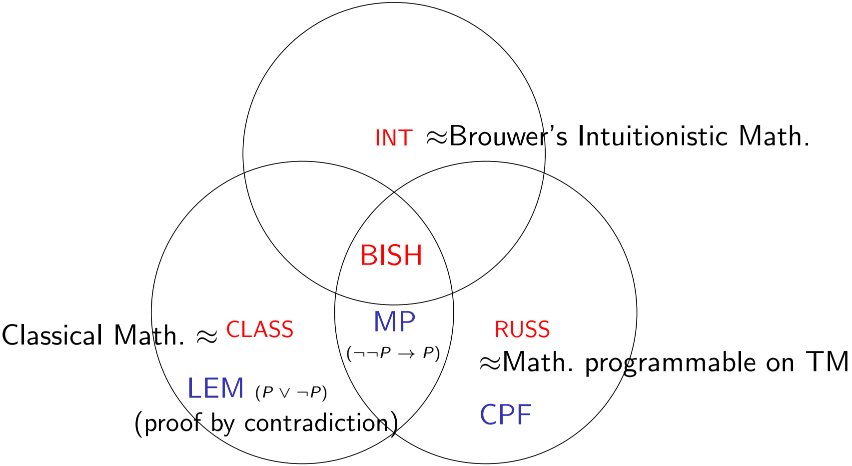
Son of a. . .


INT ≈Brouwer's Intuitionistic Math.
BISH
Classical Math. ≈ CLASS
RUSS
≈Math. programmable on TM
LEM $(P \vee \neg P)$
(proof by contradiction)

Son of a...

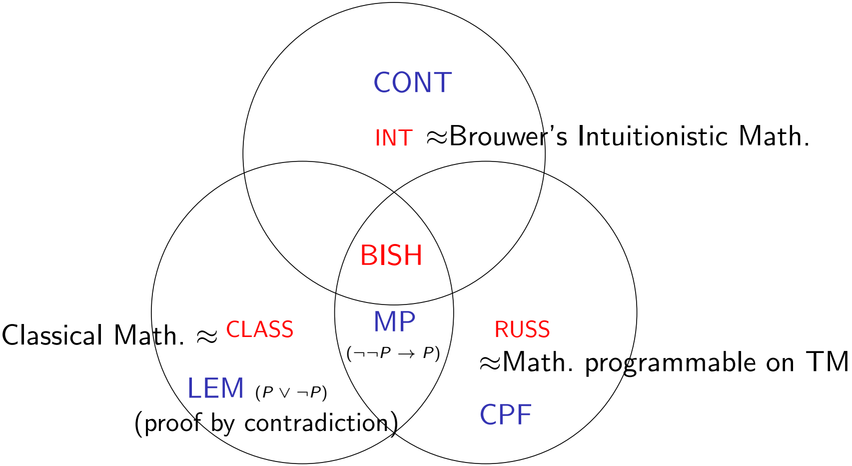

INT ≈Brouwer's Intuitionistic Math.
BISH
Classical Math. ≈ CLASS
MP
$(\neg\neg P \to P)$
RUSS
≈Math. programmable on TM
LEM $(P \vee \neg P)$
(proof by contradiction)

Son of a...

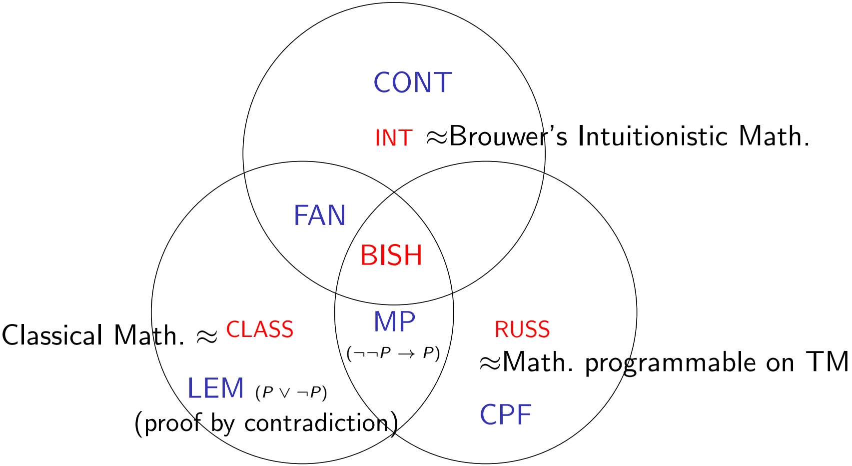


INT ≈Brouwer's Intuitionistic Math.
BISH
Classical Math. ≈ CLASS
MP
$(\neg\neg P \to P)$
RUSS
≈Math. programmable on TM
LEM $(P \vee \neg P)$
(proof by contradiction)
CPF

Son of a. . .

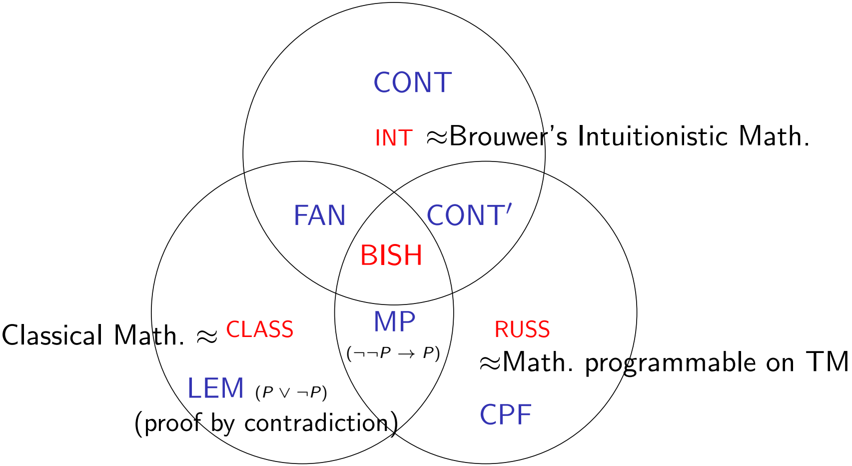

CONT
INT ≈Brouwer's Intuitionistic Math.
BISH
Classical Math. ≈ CLASS
MP
(¬¬P → P)
RUSS
≈Math. programmable on TM
LEM (P ∨ ¬P)
(proof by contradiction)
CPF

Son of a...



CONT
INT ≈Brouwer's Intuitionistic Math.
FAN
BISH
MP
$(\neg\neg P \to P)$
Classical Math. ≈ CLASS
RUSS
≈Math. programmable on TM
LEM $(P \vee \neg P)$
(proof by contradiction)
CPF

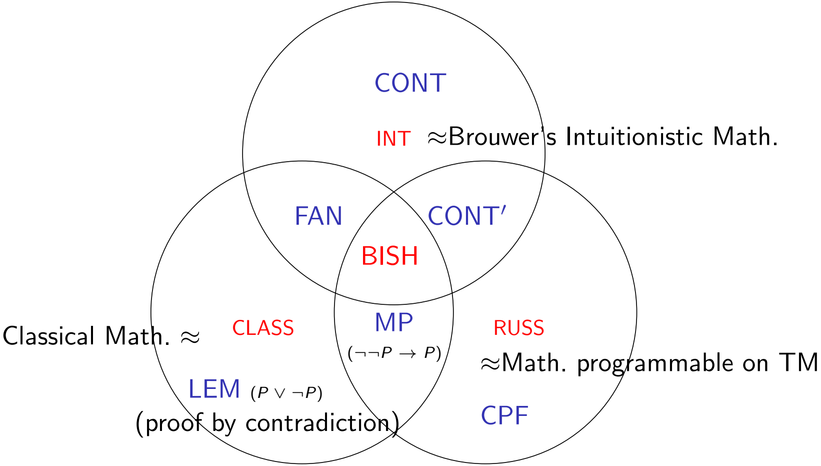
Son of a. . .



CONT
INT ≈Brouwer's Intuitionistic Math.
FAN
CONT′
BISH
Classical Math. ≈ CLASS
MP
$(\neg\neg P \to P)$
RUSS
≈Math. programmable on TM
LEM $(P \vee \neg P)$
(proof by contradiction)
CPF

Son of a…

Errett Bishop's Constructive Analysis (also 'BISH') is a constructive redevelopment of Mathematics, consistent with CLASS, RUSS and INT.



Algorithm and Proof in Constructive Analysis

Errett Bishop's Constructive Analysis (BISH) is a constructive redevelopment of Mathematics, where algorithm and proof are central.

Algorithm and Proof in Constructive Analysis

Errett Bishop's Constructive Analysis (BISH) is a constructive redevelopment of Mathematics, where algorithm and proof are central.

Definition (Logical connectives in BISH: BHK)

Algorithm and Proof in Constructive Analysis

Errett Bishop's Constructive Analysis (BISH) is a constructive redevelopment of Mathematics, where algorithm and proof are central.

Definition (Logical connectives in BISH: BHK)

1. $P \vee Q$: we have an algorithm that outputs either P or Q, together with a proof of the chosen disjunct.

Algorithm and Proof in Constructive Analysis

Errett Bishop's Constructive Analysis (BISH) is a constructive redevelopment of Mathematics, where algorithm and proof are central.

Definition (Logical connectives in BISH: BHK)

1. $P \vee Q$: we have an algorithm that outputs either P or Q, together with a proof of the chosen disjunct.
2. $P \wedge Q$: we have both a proof of P and a proof of Q.

Algorithm and Proof in Constructive Analysis

Errett Bishop's Constructive Analysis (BISH) is a constructive redevelopment of Mathematics, where algorithm and proof are central.

Definition (Logical connectives in BISH: BHK)

1. $P \vee Q$: we have an algorithm that outputs either P or Q, together with a proof of the chosen disjunct.
2. $P \wedge Q$: we have both a proof of P and a proof of Q.
3. $P \to Q$: by means of an algorithm we can convert any proof of P into a proof of Q.

Algorithm and Proof in Constructive Analysis

Errett Bishop's Constructive Analysis (BISH) is a constructive redevelopment of Mathematics, where algorithm and proof are central.

Definition (Logical connectives in BISH: BHK)

1. $P \vee Q$: we have an algorithm that outputs either P or Q, together with a proof of the chosen disjunct.
2. $P \wedge Q$: we have both a proof of P and a proof of Q.
3. $P \to Q$: by means of an algorithm we can convert any proof of P into a proof of Q.
4. $\neg P \equiv P \to (0 = 1)$.

Algorithm and Proof in Constructive Analysis

Errett Bishop's Constructive Analysis (BISH) is a constructive redevelopment of Mathematics, where algorithm and proof are central.

Definition (Logical connectives in BISH: BHK)

1. $P \vee Q$: we have an algorithm that outputs either P or Q, together with a proof of the chosen disjunct.
2. $P \wedge Q$: we have both a proof of P and a proof of Q.
3. $P \to Q$: by means of an algorithm we can convert any proof of P into a proof of Q.
4. $\neg P \equiv P \to (0 = 1)$.
5. $(\exists x) P(x)$: an algorithm computes an object x_0 such that $P(x_0)$

Algorithm and Proof in Constructive Analysis

Errett Bishop's Constructive Analysis (BISH) is a constructive redevelopment of Mathematics, where algorithm and proof are central.

Definition (Logical connectives in BISH: BHK)

1. $P \vee Q$: we have an algorithm that outputs either P or Q, together with a proof of the chosen disjunct.
2. $P \wedge Q$: we have both a proof of P and a proof of Q.
3. $P \to Q$: by means of an algorithm we can convert any proof of P into a proof of Q.
4. $\neg P \equiv P \to (0 = 1)$.
5. $(\exists x) P(x)$: an algorithm computes an object x_0 such that $P(x_0)$
6. $(\forall x \in A) P(x)$: for all x, $x \in A \to P(x)$.

From BISH to NSA

In BISH, proof and algorithm are central.

From BISH to NSA

In BISH, proof and algorithm are central.

We define a system of Nonstandard Analyis $\mathbb{NSA}$ where Transfer ($\mathbb{T}$) and Ω-invariant procedure play the same role.

From BISH to NSA

In BISH, proof and algorithm are central.

We define a system of Nonstandard Analyis $\mathbb{NSA}$ where Transfer ($\mathbb{T}$) and Ω-invariant procedure play the same role.

$\mathbb{NSA}$ is similar to $^*RCA_0$, a nonstandard version of RCA_0 (Keisler & Yokoyama).

From BISH to NSA

In BISH, proof and algorithm are central.

We define a system of Nonstandard Analyis $\mathbb{NSA}$ where Transfer ($\mathbb{T}$) and Ω-invariant procedure play the same role.

$\mathbb{NSA}$ is similar to $^*RCA_0$, a nonstandard version of RCA_0 (Keisler & Yokoyama).

Three important features:

From BISH to NSA

In BISH, proof and algorithm are central.

We define a system of Nonstandard Analyis $\mathbb{NSA}$ where Transfer ($\mathbb{T}$) and Ω-invariant procedure play the same role.

$\mathbb{NSA}$ is similar to $^*\text{RCA}_0$, a nonstandard version of RCA_0 (Keisler & Yokoyama).

Three important features:

1. No Transfer Principle, except for Δ_0.

From BISH to NSA

In BISH, proof and algorithm are central.

We define a system of Nonstandard Analyis $\mathbb{NSA}$ where Transfer ($\mathbb{T}$) and Ω-invariant procedure play the same role.

$\mathbb{NSA}$ is similar to $^*RCA_0$, a nonstandard version of RCA_0 (Keisler & Yokoyama).

Three important features:

1. No Transfer Principle, except for Δ_0.
2. No Δ_1^0-CA, but Ω-CA. (CA for Ω-invariant formulas)

From BISH to NSA

In BISH, proof and algorithm are central.

We define a system of Nonstandard Analyis $\mathbb{NSA}$ where Transfer ($\mathbb{T}$) and Ω-invariant procedure play the same role.

$\mathbb{NSA}$ is similar to $^*\mathrm{RCA}_0$, a nonstandard version of RCA_0 (Keisler & Yokoyama).

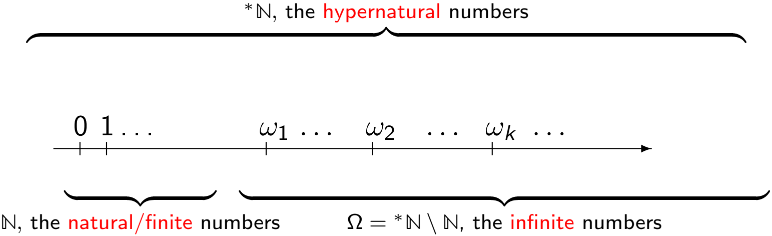
Three important features:

1. No Transfer Principle, except for Δ_0.
2. No Δ^0_1-CA, but Ω-CA. (CA for Ω-invariant formulas)
3. Levels of infinity (Stratified NSA).

Feature 3: Stratified Nonstandard Analysis

Feature 3: Stratified Nonstandard Analysis

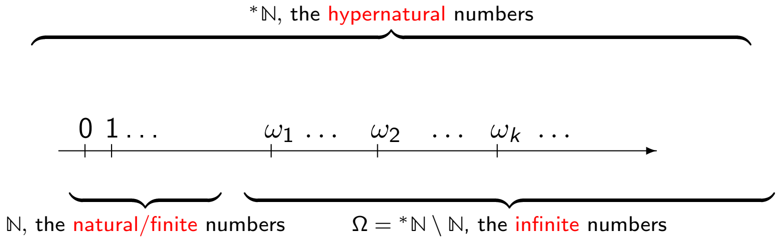
The usual picture of $^*\mathbb{N}$:



Feature 3: Stratified Nonstandard Analysis

The usual picture of ${}^*\mathbb{N}$:

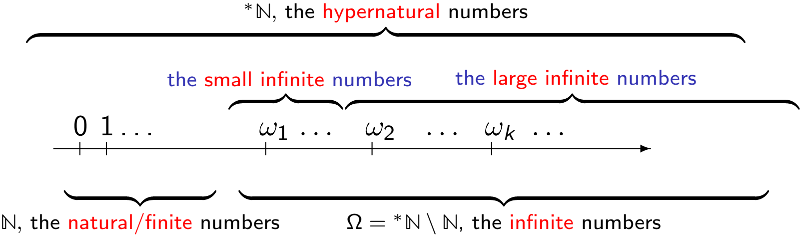
In $\mathbb{NSA}$, the infinite numbers are split into 'small' and 'large'.



Feature 3: Stratified Nonstandard Analysis

The usual picture of $^*\mathbb{N}$:

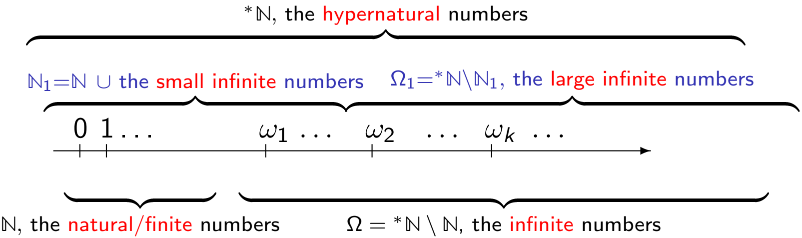
In $\mathbb{NSA}$, the infinite numbers are split into 'small' and 'large'.



Feature 3: Stratified Nonstandard Analysis

The usual picture of ${}^*\mathbb{N}$:

In $\mathbb{NSA}$, the infinite numbers are split into 'small' and 'large'.



Feature 2: Ω-invariance

Feature 2: Ω-invariance

Ω-invariance $\approx$ algorithm $\approx$ finite procedure

Definition (Ω-invariance)

For $\psi(n, m) \in \Delta_0$ and $\omega \in \Omega$, the formula $\psi(n, \omega)$ is Ω-invariant if

Feature 2: Ω-invariance

Ω-invariance $\approx$ algorithm $\approx$ finite procedure

Definition (Ω-invariance)

For $\psi(n, m) \in \Delta_0$ and $\omega \in \Omega$, the formula $\psi(n, \omega)$ is Ω-invariant if

$$(\forall n \in \mathbb{N})(\forall \omega' \in \Omega)[\psi(n, \omega) \leftrightarrow \psi(n, \omega')].$$

Feature 2: Ω-invariance

Ω-invariance $\approx$ algorithm $\approx$ finite procedure

Definition (Ω-invariance)

For $\psi(n, m) \in \Delta_0$ and $\omega \in \Omega$, the formula $\psi(n, \omega)$ is Ω-invariant if

$$(\forall n \in \mathbb{N})(\forall \omega' \in \Omega)[\psi(n, \omega) \leftrightarrow \psi(n, \omega')].$$

Note that $\psi(n, \omega)$ depends on $\omega \in \Omega$, but not on the choice of $\omega \in \Omega$.

Feature 2: Ω-invariance

Ω-invariance $\approx$ algorithm $\approx$ finite procedure

Definition (Ω-invariance)

For $\psi(n, m) \in \Delta_0$ and $\omega \in \Omega$, the formula $\psi(n, \omega)$ is Ω-invariant if

$$(\forall n \in \mathbb{N})(\forall \omega' \in \Omega)[\psi(n, \omega) \leftrightarrow \psi(n, \omega')].$$

Note that $\psi(n, \omega)$ depends on $\omega \in \Omega$, but not on the choice of $\omega \in \Omega$.

$\mathbb{NSA}$ has Ω-CA instead of Δ_1-CA.

Principle (Ω-CA)

For all Ω-invariant $\psi(n, \omega)$, we have

$$(\exists X \subset \mathbb{N})(\forall n \in \mathbb{N})(n \in X \leftrightarrow \psi(n, \omega)).$$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

Central: algorithm and proof

$A \vee B$:
an algo yields a proof of A or of B

$\mathbb{NSA}$ (based on CL)

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

Central: algorithm and proof

$A \vee B$:
an algo yields a proof of A or of B

$\mathbb{NSA}$ (based on CL)

Central: Ω-invariance and Transfer ($\mathbb{T}$)

$A \,\mathbb{V}\, B$:

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

Central: algorithm and proof

$A \vee B$:
an algo yields a proof of A or of B

$\mathbb{NSA}$ (based on CL)

Central: Ω-invariance and Transfer ($\mathbb{T}$)

$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t.

$$\psi(\vec{x},\omega) \rightarrow [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$$
$$\wedge$$
$$\neg\psi(\vec{x},\omega) \rightarrow [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

Central: algorithm and proof

$A \vee B$: an algo yields a proof of A or of B

$A \rightarrow B$: an algo converts a proof of A to a proof of B

$\mathbb{NSA}$ (based on CL)

Central: Ω-invariance and Transfer ($\mathbb{T}$)

$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t.

$$\psi(\vec{x},\omega) \rightarrow [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$$
$$\wedge$$
$$\neg\psi(\vec{x},\omega) \rightarrow [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x}, \omega)$ s.t. $\psi(\vec{x}, \omega) \rightarrow [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$ $\wedge$ $\neg\psi(\vec{x}, \omega) \rightarrow [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$
$A \rightarrow B$: an algo converts a proof of A to a proof of B	$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \rightarrow \big[B \wedge [B \in \mathbb{T}]\big]$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

Central: algorithm and proof

$A \vee B$:
an algo yields a proof of A or of B

$A \rightarrow B$: an algo converts a proof of A to a proof of B

$\mathbb{NSA}$ (based on CL)

Central: Ω-invariance and Transfer ($\mathbb{T}$)

$A \mathbin{\mathbb{V}} B$: There is Ω-invariant $\psi(\vec{x}, \omega)$ s.t.

$$\psi(\vec{x}, \omega) \rightarrow [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$$
$$\wedge$$
$$\neg\psi(\vec{x}, \omega) \rightarrow [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$$

$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \rightarrow \big[B \wedge [B \in \mathbb{T}]\big]$

'$A \in \mathbb{T}$' means 'A satisfies Transfer'.

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	$A \mathbin{\mathbb{V}} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t. $\psi(\vec{x},\omega) \to [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$ $\wedge$ $\neg\psi(\vec{x},\omega) \to [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$
$A \to B$: an algo converts a proof of A to a proof of B	$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \to \big[B \wedge [B \in \mathbb{T}]\big]$
	'$A \in \mathbb{T}$' means 'A satisfies Transfer'.
E.g. '$(\forall n \in \mathbb{N})\varphi(n) \in \mathbb{T}$'	is $[(\forall n \in \mathbb{N})\varphi(n) \to (\forall n \in {}^*\mathbb{N})\varphi(n)]$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t. $\psi(\vec{x},\omega) \rightarrow [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$ $\wedge$ $\neg\psi(\vec{x},\omega) \rightarrow [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$
$A \rightarrow B$: an algo converts a proof of A to a proof of B	$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \rightarrow \big[B \wedge [B \in \mathbb{T}]\big]$

'$A \in \mathbb{T}$' means 'A satisfies Transfer'.

E.g. '$(\forall n \in \mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\forall n \in \mathbb{N})\varphi(n) \rightarrow (\forall n \in {}^*\mathbb{N})\varphi(n)]$

E.g. '$(\exists n \in {}^*\mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\exists n \in {}^*\mathbb{N})\varphi(n) \rightarrow (\exists n \in \mathbb{N}_1)\varphi(n)]$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

Central: algorithm and proof

$A \vee B$:
an algo yields a proof of A or of B

$A \to B$: an algo converts a proof of A to a proof of B

$\neg A$: $A \to (0 = 1)$

$\mathbb{NSA}$ (based on CL)

Central: Ω-invariance and Transfer ($\mathbb{T}$)

$A \mathbin{\mathbb{V}} B$: There is Ω-invariant $\psi(\vec{x}, \omega)$ s.t.

$$\psi(\vec{x},\omega) \to [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$$
$$\wedge$$
$$\neg\psi(\vec{x},\omega) \to [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$$

$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \to \big[B \wedge [B \in \mathbb{T}]\big]$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t. $\psi(\vec{x},\omega) \rightarrow [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$ $\wedge$ $\neg\psi(\vec{x},\omega) \rightarrow [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$
$A \rightarrow B$: an algo converts a proof of A to a proof of B	$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \rightarrow \big[B \wedge [B \in \mathbb{T}]\big]$
$\neg A$: $A \rightarrow (0=1)$	$\sim A$: $A \Rrightarrow (0=1)$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

Central: algorithm and proof

$A \vee B$: an algo yields a proof of A or of B

$A \to B$: an algo converts a proof of A to a proof of B

$\neg A$: $A \to (0 = 1)$

$(\exists x)A(x)$: an algo computes x_0 such that $A(x_0)$

$\mathbb{NSA}$ (based on CL)

Central: Ω-invariance and Transfer ($\mathbb{T}$)

$A \mathbin{\mathbb{V}} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t.

$$\psi(\vec{x},\omega) \to [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]] \\ \wedge \\ \neg\psi(\vec{x},\omega) \to [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$$

$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \to \big[B \wedge [B \in \mathbb{T}]\big]$

$\sim A$: $A \Rrightarrow (0 = 1)$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

Central: algorithm and proof

$A \vee B$: an algo yields a proof of A or of B

$A \to B$: an algo converts a proof of A to a proof of B

$\neg A$: $A \to (0 = 1)$

$(\exists x)A(x)$: an algo computes x_0 such that $A(x_0)$

$\mathbb{NSA}$ (based on CL)

Central: Ω-invariance and Transfer ($\mathbb{T}$)

$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t.

$$\psi(\vec{x},\omega) \to [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$$
$$\wedge$$
$$\neg\psi(\vec{x},\omega) \to [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$$

$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \to \big[B \wedge [B \in \mathbb{T}]\big]$

$\sim A$: $A \Rrightarrow (0 = 1)$

$(\exists x)A(x)$: "an Ω-inv. proc. computes x_0 such that $A(x_0)$"

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x}, \omega)$ s.t. $\psi(\vec{x}, \omega) \rightarrow [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$ $\wedge$ $\neg\psi(\vec{x}, \omega) \rightarrow [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$
$A \rightarrow B$: an algo converts a proof of A to a proof of B	$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \rightarrow \big[B \wedge [B \in \mathbb{T}]\big]$
$\neg A$: $A \rightarrow (0 = 1)$	$\sim A$: $A \Rrightarrow (0 = 1)$
$(\exists x)A(x)$: an algo computes x_0 such that $A(x_0)$	$(\exists x)A(x)$: "an Ω-inv. proc. computes x_0 such that $A(x_0)$"
	$\sim[(\forall n \in \mathbb{N})A(n)]$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	$A \mathbin{\mathbb{V}} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t. $\psi(\vec{x},\omega) \to [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$ $\wedge$ $\neg\psi(\vec{x},\omega) \to [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$
$A \to B$: an algo converts a proof of A to a proof of B	$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \to \big[B \wedge [B \in \mathbb{T}]\big]$
$\neg A$: $A \to (0=1)$	$\sim A$: $A \Rrightarrow (0=1)$
$(\exists x)A(x)$: an algo computes x_0 such that $A(x_0)$	$(\exists x)A(x)$: "an Ω-inv. proc. computes x_0 such that $A(x_0)$"

$\sim[(\forall n \in \mathbb{N})A(n)] \equiv (\exists n \in \mathbb{N}_1)\sim A(n)$

WEAKER than $(\exists n \in \mathbb{N})\sim A(n)$.

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)

Central: algorithm and proof

$A \vee B$: an algo yields a proof of A or of B

$A \to B$: an algo converts a proof of A to a proof of B

$\neg A$: $A \to (0 = 1)$

$(\exists x)A(x)$: an algo computes x_0 such that $A(x_0)$

$\neg[(\forall n \in \mathbb{N})A(n)]$ is WEAKER than $(\exists n \in \mathbb{N})\neg A(n)$.

$\mathbb{NSA}$ (based on CL)

Central: Ω-invariance and Transfer ($\mathbb{T}$)

$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x}, \omega)$ s.t.

$$\psi(\vec{x}, \omega) \to [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$$
$$\wedge$$
$$\neg\psi(\vec{x}, \omega) \to [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$$

$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \to \big[B \wedge [B \in \mathbb{T}]\big]$

$\sim A$: $A \Rrightarrow (0 = 1)$

$(\exists x)A(x)$: "an Ω-inv. proc. computes x_0 such that $A(x_0)$"

$\sim[(\forall n \in \mathbb{N})A(n)] \equiv (\exists n \in \mathbb{N}_1)\sim A(n)$

WEAKER than $(\exists n \in \mathbb{N})\sim A(n)$.

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	$A \mathbin{\mathbb{V}} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t. $\psi(\vec{x},\omega) \to [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$ $\wedge$ $\neg\psi(\vec{x},\omega) \to [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$
$A \to B$: an algo converts a proof of A to a proof of B	$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \to \big[B \wedge [B \in \mathbb{T}]\big]$
$\neg A$: $A \to (0 = 1)$	$\sim A$: $A \Rrightarrow (0 = 1)$
$(\exists x)A(x)$: an algo computes x_0 such that $A(x_0)$	$(\exists x)A(x)$: "an Ω-inv. proc. computes x_0 such that $A(x_0)$"

We know: If BISH $\vdash X$ then $X \not\to$ LPO, LLPO, MP, ... (princ. rejected in BISH)

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x},\omega)$ s.t. $\psi(\vec{x},\omega) \to [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$ $\wedge$ $\neg\psi(\vec{x},\omega) \to [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$
$A \to B$: an algo converts a proof of A to a proof of B	$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \to \big[B \wedge [B \in \mathbb{T}]\big]$
$\neg A$: $A \to (0 = 1)$	$\sim A$: $A \Rrightarrow (0 = 1)$
$(\exists x)A(x)$: an algo computes x_0 such that $A(x_0)$	$(\exists x)A(x)$: "an Ω-inv. proc. computes x_0 such that $A(x_0)$"

We know: If BISH $\vdash X$ then $X \not\to$ LPO, LLPO, MP, ... (princ. rejected in BISH)

We show: If $\mathbb{NSA} \vdash Y$ then $Y \not\Rrightarrow \mathbb{LPO}, \mathbb{LLPO}, \mathbb{MP}, \dots$

The translation $\mathbb{B}$ from BISH to $\mathbb{NSA}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
Central: algorithm and proof	Central: Ω-invariance and Transfer ($\mathbb{T}$)
$A \vee B$: an algo yields a proof of A or of B	$A \mathbb{V} B$: There is Ω-invariant $\psi(\vec{x}, \omega)$ s.t. $\psi(\vec{x}, \omega) \rightarrow [A(\vec{x}) \wedge [A(\vec{x}) \in \mathbb{T}]]$ $\wedge$ $\neg\psi(\vec{x}, \omega) \rightarrow [B(\vec{x}) \wedge [B(\vec{x}) \in \mathbb{T}]]$
$A \rightarrow B$: an algo converts a proof of A to a proof of B	$A \Rrightarrow B$: $\big[A \wedge [A \in \mathbb{T}]\big] \rightarrow \big[B \wedge [B \in \mathbb{T}]\big]$
$\neg A$: $A \rightarrow (0 = 1)$	$\sim A$: $A \Rrightarrow (0 = 1)$
$(\exists x)A(x)$: an algo computes x_0 such that $A(x_0)$	$(\exists x)A(x)$: "an Ω-inv. proc. computes x_0 such that $A(x_0)$"

We know: If BISH $\vdash X$ then $X \not\rightarrow$ LPO, LLPO, MP, ... (princ. rejected in BISH)

We show: If $\mathbb{NSA} \vdash Y$ then $Y \not\Rrightarrow \mathbb{LPO}, \mathbb{LLPO}, \mathbb{MP}, \ldots$ (e.g. $\mathbb{LPO}$ is $\mathbb{B}$(LPO), unprovable in $\mathbb{NSA}$

Constructive Reverse Mathematics under $\mathbb{B}$

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	
LPO: For $P \in \Sigma_1$, $P \vee \neg P$	
$\updownarrow$	

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK) | $\mathbb{NSA}$ (based on CL)

non-constructive/non-algorithmic

LPO: For $P \in \Sigma_1$, $P \vee \neg P$

$\updownarrow$

LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$

$\updownarrow$

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK) | $\mathbb{NSA}$ (based on CL)

non-constructive/non-algorithmic

LPO: For $P \in \Sigma_1$, $P \vee \neg P$

$\updownarrow$

LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$

$\updownarrow$

MCT: monotone convergence thm

$\updownarrow$

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)

non-constructive/non-algorithmic

LPO: For $P \in \Sigma_1$, $P \vee \neg P$

$\updownarrow$

LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$

$\updownarrow$

MCT: monotone convergence thm

$\updownarrow$

CIT: Cantor intersection thm

$\mathbb{NSA}$ (based on CL)

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)

non-constructive/non-algorithmic

LPO: For $P \in \Sigma_1$, $P \vee \neg P$

$\updownarrow$

LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$

$\updownarrow$

MCT: monotone convergence thm

$\updownarrow$

CIT: Cantor intersection thm

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)

non-constructive/non-algorithmic

LPO: For $P \in \Sigma_1$, $P \vee \neg P$

$\updownarrow$

LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$

$\updownarrow$

MCT: monotone convergence thm

$\updownarrow$

CIT: Cantor intersection thm

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbb{V} \sim P$

$\Longleftrightarrow$

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
LPO: For $P \in \Sigma_1$, $P \vee \neg P$	$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbb{V} \sim P$
$\updownarrow$	$\Longleftrightarrow$
LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$	$\mathbb{LPR}$: $(\forall x \in \mathbb{R})(x > 0 \mathbb{V} \sim(x > 0))$
$\updownarrow$	$\Longleftrightarrow$
MCT: monotone convergence thm	
$\updownarrow$	
CIT: Cantor intersection thm	

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)

non-constructive/non-algorithmic

LPO: For $P \in \Sigma_1$, $P \vee \neg P$

$\updownarrow$

LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$

$\updownarrow$

MCT: monotone convergence thm

$\updownarrow$

CIT: Cantor intersection thm

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbb{V} {\sim}P$

$\Longleftrightarrow$

$\mathbb{LPR}$: $(\forall x \in \mathbb{R})(x > 0 \mathbb{V} {\sim}(x > 0))$

$\Longleftrightarrow$

$\mathbb{MCT}$: monotone convergence thm

$\Longleftrightarrow$

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
LPO: For $P \in \Sigma_1$, $P \vee \neg P$	$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbb{V} \sim P$
$\updownarrow$	$\Longleftrightarrow$
LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$	$\mathbb{LPR}$: $(\forall x \in \mathbb{R})(x > 0 \mathbb{V} \sim(x > 0))$
$\updownarrow$	$\Longleftrightarrow$
MCT: monotone convergence thm	$\mathbb{MCT}$: monotone convergence thm
$\updownarrow$	$\Longleftrightarrow$
CIT: Cantor intersection thm	$\mathbb{CIT}$: Cantor intersection thm

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
LPO: For $P \in \Sigma_1$, $P \vee \neg P$	$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbb{V} \sim P$
$\updownarrow$	$\Longleftrightarrow$
LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$	$\mathbb{LPR}$: $(\forall x \in \mathbb{R})(x > 0 \mathbb{V} \sim(x > 0))$
$\updownarrow$	$\Longleftrightarrow$
MCT: monotone convergence thm	$\mathbb{MCT}$: monotone convergence thm
$\updownarrow$ (limit computed by algo)	$\Longleftrightarrow$
CIT: Cantor intersection thm	$\mathbb{CIT}$: Cantor intersection thm

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)		$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic		non-Ω-invariant
LPO: For $P \in \Sigma_1$, $P \vee \neg P$		$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbin{\mathbb{V}} \sim P$
$\updownarrow$		$\Longleftrightarrow$
LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$		$\mathbb{LPR}$: $(\forall x \in \mathbb{R})(x > 0 \mathbin{\mathbb{V}} \sim(x > 0))$
$\updownarrow$		$\Longleftrightarrow$
MCT: monotone convergence thm		$\mathbb{MCT}$: monotone convergence thm
$\updownarrow$ (limit computed by algo)	$\Longleftrightarrow$	(limit computed by Ω-inv. proc.)
CIT: Cantor intersection thm		$\mathbb{CIT}$: Cantor intersection thm

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
LPO: For $P \in \Sigma_1$, $P \vee \neg P$	$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbb{V} \sim P$
$\updownarrow$	$\Longleftrightarrow$
LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$	$\mathbb{LPR}$: $(\forall x \in \mathbb{R})(x > 0 \mathbb{V} \sim(x > 0))$
$\updownarrow$	$\Longleftrightarrow$
MCT: monotone convergence thm	$\mathbb{MCT}$: monotone convergence thm
$\updownarrow$ (limit computed by algo)	$\Longleftrightarrow$ (limit computed by Ω-inv. proc.)
CIT: Cantor intersection thm	$\mathbb{CIT}$: Cantor intersection thm
(point in intersection computed by algo)	

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
LPO: For $P \in \Sigma_1$, $P \vee \neg P$	$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbb{V} \sim P$
$\updownarrow$	$\Longleftrightarrow$
LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$	$\mathbb{LPR}$: $(\forall x \in \mathbb{R})(x > 0 \mathbb{V} \sim(x > 0))$
$\updownarrow$	$\Longleftrightarrow$
MCT: monotone convergence thm	$\mathbb{MCT}$: monotone convergence thm
$\updownarrow$ (limit computed by algo) $\Longleftrightarrow$	(limit computed by Ω-inv. proc.)
CIT: Cantor intersection thm	$\mathbb{CIT}$: Cantor intersection thm

(point in intersection computed by algo)

(point in intersection computed by Ω-inv. proc.)

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)

non-constructive/non-algorithmic

LPO: For $P \in \Sigma_1$, $P \vee \neg P$

$\updownarrow$

LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$

$\updownarrow$

MCT: monotone convergence thm

$\updownarrow$ (limit computed by algo)

CIT: Cantor intersection thm

$\Longleftrightarrow$

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbb{V} \sim P$

$\Longleftrightarrow$

$\mathbb{LPR}$: $(\forall x \in \mathbb{R})(x > 0 \mathbb{V} \sim(x > 0))$

$\Longleftrightarrow$

$\mathbb{MCT}$: monotone convergence thm

$\Longleftrightarrow$ (limit computed by Ω-inv. proc.)

$\mathbb{CIT}$: Cantor intersection thm

$\Longleftrightarrow$

Universal Transfer: For all $\varphi \in \Delta_0$

$(\forall n \in \mathbb{N})\varphi(n) \rightarrow (\forall n \in {}^*\mathbb{N})\varphi(n)$

Constructive Reverse Mathematics under $\mathbb{B}$

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
LPO: For $P \in \Sigma_1$, $P \vee \neg P$	$\mathbb{LPO}$: For $P \in \Sigma_1$, $P \mathbb{V} \sim P$
$\updownarrow$	$\Longleftrightarrow$
LPR: $(\forall x \in \mathbb{R})(x > 0 \vee \neg(x > 0))$	$\mathbb{LPR}$: $(\forall x \in \mathbb{R})(x > 0 \mathbb{V} \sim (x > 0))$
$\updownarrow$	$\Longleftrightarrow$
MCT: monotone convergence thm	$\mathbb{MCT}$: monotone convergence thm
$\updownarrow$ (limit computed by algo) $\Longleftrightarrow$	(limit computed by Ω-inv. proc.)
CIT: Cantor intersection thm	$\mathbb{CIT}$: Cantor intersection thm
	$\Longleftrightarrow$
	Universal Transfer: For all $\varphi \in \Delta_0$
	$(\forall n \in \mathbb{N})\varphi(n) \rightarrow (\forall n \in {}^*\mathbb{N})\varphi(n)$

$\mathbb{NSA}$ does prove $(\forall \delta \in \mathbb{R})\big[\delta > 0 \Rrightarrow (x > 0) \mathbb{V} (x < \delta)\big]$.

BISH does prove $(\forall \delta \in \mathbb{R})\big[\delta > 0 \rightarrow (x > 0) \vee (x < \delta)\big]$.

Constructive Reverse Mathematics under $\mathbb{B}$ II

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

$\mathbb{NSA}$ (based on CL)

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$

$\updownarrow$

$\mathbb{NSA}$ (based on CL)

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

$\mathbb{NSA}$ (based on CL)

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

NIL

$(\forall x, y \in \mathbb{R})(xy = 0 \to x = 0 \vee y = 0)$

$\updownarrow$

$\mathbb{NSA}$ (based on CL)

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

NIL

$(\forall x, y \in \mathbb{R})(xy = 0 \to x = 0 \vee y = 0)$

$\updownarrow$

IVT: Intermediate value theorem

$\mathbb{NSA}$ (based on CL)

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \rightarrow \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

NIL

$(\forall x, y \in \mathbb{R})(xy = 0 \rightarrow x = 0 \vee y = 0)$

$\updownarrow$

IVT: Intermediate value theorem

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

NIL

$(\forall x, y \in \mathbb{R})(xy = 0 \to x = 0 \vee y = 0)$

$\updownarrow$

IVT: Intermediate value theorem

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{LLPO}$

For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \mathbb{V} \sim Q$

$\Longleftrightarrow$

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

NIL

$(\forall x, y \in \mathbb{R})(xy = 0 \to x = 0 \vee y = 0)$

$\updownarrow$

IVT: Intermediate value theorem

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{LLPO}$

For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \Vvee \sim Q$

$\Longleftrightarrow$

$\mathbb{LLPR}$: $(\forall x \in \mathbb{R})(x \geq 0 \Vvee x \leq 0)$

$\Longleftrightarrow$

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \rightarrow \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

NIL

$(\forall x, y \in \mathbb{R})(xy = 0 \rightarrow x = 0 \vee y = 0)$

$\updownarrow$

IVT: Intermediate value theorem

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{LLPO}$

For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \mathbb{V} \sim Q$

$\Longleftrightarrow$

$\mathbb{LLPR}$: $(\forall x \in \mathbb{R})(x \geq 0 \mathbb{V} x \leq 0)$

$\Longleftrightarrow$

$\mathbb{NIL}$

$(\forall x, y \in \mathbb{R})(xy = 0 \Rrightarrow x = 0 \mathbb{V} y = 0)$

$\Longleftrightarrow$

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

NIL

$(\forall x, y \in \mathbb{R})(xy = 0 \to x = 0 \vee y = 0)$

$\updownarrow$

IVT: Intermediate value theorem

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{LLPO}$

For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \mathbb{V} \sim Q$

$\Longleftrightarrow$

$\mathbb{LLPR}$: $(\forall x \in \mathbb{R})(x \geq 0 \mathbb{V} x \leq 0)$

$\Longleftrightarrow$

$\mathbb{NIL}$

$(\forall x, y \in \mathbb{R})(xy = 0 \Rrightarrow x = 0 \mathbb{V} y = 0)$

$\Longleftrightarrow$

$\mathbb{IVT}$: Intermediate value theorem

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

NIL

$(\forall x, y \in \mathbb{R})(xy = 0 \to x = 0 \vee y = 0)$

$\updownarrow$

IVT: Intermediate value theorem
(int. value computed by algo)

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{LLPO}$

For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \veebar \sim Q$

$\Longleftrightarrow$

$\mathbb{LLPR}$: $(\forall x \in \mathbb{R})(x \geq 0 \veebar x \leq 0)$

$\Longleftrightarrow$

$\mathbb{NIL}$

$(\forall x, y \in \mathbb{R})(xy = 0 \Rrightarrow x = 0 \veebar y = 0)$

$\Longleftrightarrow$

$\mathbb{IVT}$: Intermediate value theorem

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK) non-constructive/non-algorithmic	$\mathbb{NSA}$ (based on CL) non-Ω-invariant
LLPO	$\mathbb{LLPO}$
For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \rightarrow \neg P \vee \neg Q$	For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \mathbb{V} \sim Q$
$\updownarrow$	$\Longleftrightarrow$
LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$	$\mathbb{LLPR}$: $(\forall x \in \mathbb{R})(x \geq 0 \mathbb{V} x \leq 0)$
$\updownarrow$	$\Longleftrightarrow$
NIL	$\mathbb{NIL}$
$(\forall x, y \in \mathbb{R})(xy = 0 \rightarrow x = 0 \vee y = 0)$	$(\forall x, y \in \mathbb{R})(xy = 0 \Rrightarrow x = 0 \mathbb{V} y = 0)$
$\updownarrow$	$\Longleftrightarrow$
IVT: Intermediate value theorem (int. value computed by algo)	$\mathbb{IVT}$: Intermediate value theorem (int. value computed by Ω-inv. proc.)

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
LLPO	$\mathbb{LLPO}$
For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$	For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \,\mathbb{V} \sim Q$
$\updownarrow$	$\Longleftrightarrow$
LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$	$\mathbb{LLPR}$: $(\forall x \in \mathbb{R})(x \geq 0 \,\mathbb{V}\, x \leq 0)$
$\updownarrow$	$\Longleftrightarrow$
NIL	$\mathbb{NIL}$
$(\forall x, y \in \mathbb{R})(xy = 0 \to x = 0 \vee y = 0)$	$(\forall x, y \in \mathbb{R})(xy = 0 \Rrightarrow x = 0 \,\mathbb{V}\, y = 0)$
$\updownarrow$	$\Longleftrightarrow$
IVT: Intermediate value theorem	$\mathbb{IVT}$: Intermediate value theorem
$\updownarrow$ (int. value computed by algo)	(int. value computed by Ω-inv. proc.)
WKL	$\Longleftrightarrow$ $\mathbb{WKL}$

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
LLPO	$\mathbb{LLPO}$
For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$	For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \mathbb{V} \sim Q$
$\updownarrow$	$\Longleftrightarrow$
LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$	$\mathbb{LLPR}$: $(\forall x \in \mathbb{R})(x \geq 0 \mathbb{V} x \leq 0)$
$\updownarrow$	$\Longleftrightarrow$
NIL	$\mathbb{NIL}$
$(\forall x, y \in \mathbb{R})(xy = 0 \to x = 0 \vee y = 0)$	$(\forall x, y \in \mathbb{R})(xy = 0 \Rrightarrow x = 0 \mathbb{V} y = 0)$
$\updownarrow$	$\Longleftrightarrow$
IVT: Intermediate value theorem	$\mathbb{IVT}$: Intermediate value theorem
$\updownarrow$ (int. value computed by algo)	(int. value computed by Ω-inv. proc.)
WKL	$\Longleftrightarrow$ $\mathbb{WKL}$ $\Longleftrightarrow$ $\mathbb{V}$-Transfer

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK)

non-constructive/non-algorithmic

LLPO

For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \to \neg P \vee \neg Q$

$\updownarrow$

LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$

$\updownarrow$

NIL

$(\forall x, y \in \mathbb{R})(xy = 0 \to x = 0 \vee y = 0)$

$\updownarrow$

IVT: Intermediate value theorem

$\updownarrow$ (int. value computed by algo)

WKL

Axioms of $\mathbb{R}$: $\neg(x > 0 \wedge x < 0)$

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{LLPO}$

For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \,\mathbb{V}\, \sim Q$

$\Longleftrightarrow$

$\mathbb{LLPR}$: $(\forall x \in \mathbb{R})(x \geq 0 \,\mathbb{V}\, x \leq 0)$

$\Longleftrightarrow$

$\mathbb{NIL}$

$(\forall x, y \in \mathbb{R})(xy = 0 \Rrightarrow x = 0 \,\mathbb{V}\, y = 0)$

$\Longleftrightarrow$

$\mathbb{IVT}$: Intermediate value theorem

(int. value computed by Ω-inv. proc.)

$\Longleftrightarrow$ $\mathbb{WKL}$ $\Longleftrightarrow$ $\mathbb{V}$-Transfer

Constructive Reverse Mathematics under $\mathbb{B}$ II

BISH (based on BHK) non-constructive/non-algorithmic	$\mathbb{NSA}$ (based on CL) non-Ω-invariant
LLPO	$\mathbb{LLPO}$
For $P, Q \in \Sigma_1$, $\neg(P \wedge Q) \rightarrow \neg P \vee \neg Q$	For $P, Q \in \Sigma_1$, $\sim(P \wedge Q) \Rrightarrow \sim P \mathbb{V} \sim Q$
$\updownarrow$	$\Longleftrightarrow$
LLPR: $(\forall x \in \mathbb{R})(x \geq 0 \vee x \leq 0)$	$\mathbb{LLPR}$: $(\forall x \in \mathbb{R})(x \geq 0 \mathbb{V} x \leq 0)$
$\updownarrow$	$\Longleftrightarrow$
NIL	$\mathbb{NIL}$
$(\forall x, y \in \mathbb{R})(xy = 0 \rightarrow x = 0 \vee y = 0)$	$(\forall x, y \in \mathbb{R})(xy = 0 \Rrightarrow x = 0 \mathbb{V} y = 0)$
$\updownarrow$	$\Longleftrightarrow$
IVT: Intermediate value theorem (int. value computed by algo)	$\mathbb{IVT}$: Intermediate value theorem (int. value computed by Ω-inv. proc.)
$\updownarrow$ WKL	$\Longleftrightarrow$ $\mathbb{WKL}$ $\Longleftrightarrow$ $\vee$-Transfer
Axioms of $\mathbb{R}$: $\neg(x > 0 \wedge x < 0)$	Axioms of $\mathbb{R}$: $\sim(x > 0 \wedge x < 0)$

Constructive Reverse Mathematics under $\mathbb{B}$ III

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)

non-constructive/non-algorithmic

MP: For $P \in \Sigma_1$, $\neg\neg P \to P$

$\updownarrow$

$\mathbb{NSA}$ (based on CL)

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)

non-constructive/non-algorithmic

MP: For $P \in \Sigma_1$, $\neg\neg P \to P$

$\updownarrow$

MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$

$\updownarrow$

$\mathbb{NSA}$ (based on CL)

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic MP: For $P \in \Sigma_1$, $\neg\neg P \to P$ $\updownarrow$ MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$ $\updownarrow$ EXT: the extensionality theorem	

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \to P$ $\updownarrow$ MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$ $\updownarrow$ EXT: the extensionality theorem	

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \to P$	$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rrightarrow P$
$\updownarrow$	$\Longleftrightarrow$
MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$	
$\updownarrow$	
EXT: the extensionality theorem	

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \to P$	$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rrightarrow P$
$\updownarrow$	$\Longleftrightarrow$
MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$	$\mathbb{MPR}$: $(\forall x \in \mathbb{R})(\sim\sim(x > 0) \Rrightarrow x > 0)$
$\updownarrow$	$\Longleftrightarrow$
EXT: the extensionality theorem	

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK) non-constructive/non-algorithmic	$\mathbb{NSA}$ (based on CL) non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \rightarrow P$	$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rrightarrow P$
$\updownarrow$	$\Longleftrightarrow$
MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \rightarrow x > 0)$	$\mathbb{MPR}$: $(\forall x \in \mathbb{R})(\sim\sim(x > 0) \Rrightarrow x > 0)$
$\updownarrow$	$\Longleftrightarrow$
EXT: the extensionality theorem	$\mathbb{EXT}$: the extensionality theorem

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \to P$	$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rightarrow P$
$\updownarrow$	$\Longleftrightarrow$
MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$	$\mathbb{MPR}$: $(\forall x \in \mathbb{R})(\sim\sim(x > 0) \Rightarrow x > 0)$
$\updownarrow$	$\Longleftrightarrow$
EXT: the extensionality theorem	$\mathbb{EXT}$: the extensionality theorem
WLPO: For $P \in \Sigma_1$, $\neg\neg P \vee \neg P$	
$\updownarrow$	

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK) non-constructive/non-algorithmic	$\mathbb{NSA}$ (based on CL) non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \rightarrow P$	$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rrightarrow P$
$\updownarrow$	$\Longleftrightarrow$
MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \rightarrow x > 0)$	$\mathbb{MPR}$: $(\forall x \in \mathbb{R})(\sim\sim(x > 0) \Rrightarrow x > 0)$
$\updownarrow$	$\Longleftrightarrow$
EXT: the extensionality theorem	$\mathbb{EXT}$: the extensionality theorem
WLPO: For $P \in \Sigma_1$, $\neg\neg P \vee \neg P$	
$\updownarrow$	
WLPR: $(\forall x \in \mathbb{R})\big[\neg\neg(x > 0) \vee \neg(x > 0)\big]$	
$\updownarrow$	

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK) non-constructive/non-algorithmic	$\mathbb{NSA}$ (based on CL) non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \to P$	$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rrightarrow P$
$\updownarrow$	$\Longleftrightarrow$
MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$	$\mathbb{MPR}$: $(\forall x \in \mathbb{R})(\sim\sim(x > 0) \Rrightarrow x > 0)$
$\updownarrow$	$\Longleftrightarrow$
EXT: the extensionality theorem	$\mathbb{EXT}$: the extensionality theorem
WLPO: For $P \in \Sigma_1$, $\neg\neg P \vee \neg P$	
$\updownarrow$	
WLPR: $(\forall x \in \mathbb{R})\big[\neg\neg(x > 0) \vee \neg(x > 0)\big]$	
$\updownarrow$	
DISC: A discontinuous $2^{\mathbb{N}} \to \mathbb{N}$-function exists.	

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK) non-constructive/non-algorithmic	$\mathbb{NSA}$ (based on CL) non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \to P$	$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rrightarrow P$
$\updownarrow$	$\Longleftrightarrow$
MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$	$\mathbb{MPR}$: $(\forall x \in \mathbb{R})(\sim\sim(x > 0) \Rrightarrow x > 0)$
$\updownarrow$	$\Longleftrightarrow$
EXT: the extensionality theorem	$\mathbb{EXT}$: the extensionality theorem
WLPO: For $P \in \Sigma_1$, $\neg\neg P \vee \neg P$	$\mathbb{WLPO}$: For $P \in \Sigma_1$, $\sim\sim P \mathbb{V} \sim P$
$\updownarrow$	$\Longleftrightarrow$
WLPR: $(\forall x \in \mathbb{R})\left[\neg\neg(x > 0) \vee \neg(x > 0)\right]$	
$\updownarrow$	
DISC: A discontinuous $2^{\mathbb{N}} \to \mathbb{N}$-function exists.	

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)

non-constructive/non-algorithmic

MP: For $P \in \Sigma_1$, $\neg\neg P \rightarrow P$

$\updownarrow$

MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \rightarrow x > 0)$

$\updownarrow$

EXT: the extensionality theorem

WLPO: For $P \in \Sigma_1$, $\neg\neg P \vee \neg P$

$\updownarrow$

WLPR: $(\forall x \in \mathbb{R})\big[\neg\neg(x > 0) \vee \neg(x > 0)\big]$

$\updownarrow$

DISC:

A discontinuous $2^{\mathbb{N}} \rightarrow \mathbb{N}$-function exists.

$\mathbb{NSA}$ (based on CL)

non-Ω-invariant

$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rrightarrow P$

$\Longleftrightarrow$

$\mathbb{MPR}$: $(\forall x \in \mathbb{R})(\sim\sim(x > 0) \Rrightarrow x > 0)$

$\Longleftrightarrow$

$\mathbb{EXT}$: the extensionality theorem

$\mathbb{WLPO}$: For $P \in \Sigma_1$, $\sim\sim P \veebar \sim P$

$\Longleftrightarrow$

$\mathbb{WLPR}$: $(\forall x \in \mathbb{R})\big[\sim\sim(x > 0) \veebar \sim(x > 0)\big]$

$\Longleftrightarrow$

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \to P$	$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rrightarrow P$
$\updownarrow$	$\Longleftrightarrow$
MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$	$\mathbb{MPR}$: $(\forall x \in \mathbb{R})(\sim\sim(x > 0) \Rrightarrow x > 0)$
$\updownarrow$	$\Longleftrightarrow$
EXT: the extensionality theorem	$\mathbb{EXT}$: the extensionality theorem
WLPO: For $P \in \Sigma_1$, $\neg\neg P \vee \neg P$	$\mathbb{WLPO}$: For $P \in \Sigma_1$, $\sim\sim P \mathbb{V} \sim P$
$\updownarrow$	$\Longleftrightarrow$
WLPR: $(\forall x \in \mathbb{R})\big[\neg\neg(x > 0) \vee \neg(x > 0)\big]$	$\mathbb{WLPR}$: $(\forall x \in \mathbb{R})\big[\sim\sim(x > 0) \mathbb{V} \sim(x > 0)\big]$
$\updownarrow$	$\Longleftrightarrow$
DISC: A discontinuous $2^{\mathbb{N}} \to \mathbb{N}$-function exists.	$\mathbb{DISC}$: A discontinuous $2^{\mathbb{N}} \to \mathbb{N}$-function exists.

Constructive Reverse Mathematics under $\mathbb{B}$ III

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
non-constructive/non-algorithmic	non-Ω-invariant
MP: For $P \in \Sigma_1$, $\neg\neg P \to P$	$\mathbb{MP}$: For $P \in \Sigma_1$, $\sim\sim P \Rrightarrow P$
$\updownarrow$	$\Longleftrightarrow$
MPR: $(\forall x \in \mathbb{R})(\neg\neg(x > 0) \to x > 0)$	$\mathbb{MPR}$: $(\forall x \in \mathbb{R})(\sim\sim(x > 0) \Rrightarrow x > 0)$
$\updownarrow$	$\Longleftrightarrow$
EXT: the extensionality theorem	$\mathbb{EXT}$: the extensionality theorem
WLPO: For $P \in \Sigma_1$, $\neg\neg P \vee \neg P$	$\mathbb{WLPO}$: For $P \in \Sigma_1$, $\sim\sim P \mathbin{\mathbb{V}} \sim P$
$\updownarrow$	$\Longleftrightarrow$
WLPR: $(\forall x \in \mathbb{R})\big[\neg\neg(x > 0) \vee \neg(x > 0)\big]$	$\mathbb{WLPR}$: $(\forall x \in \mathbb{R})\big[\sim\sim(x > 0) \mathbin{\mathbb{V}} \sim(x > 0)\big]$
$\updownarrow$	$\Longleftrightarrow$
DISC: A discontinuous $2^{\mathbb{N}} \to \mathbb{N}$-function exists.	$\mathbb{DISC}$: A discontinuous $2^{\mathbb{N}} \to \mathbb{N}$-function exists.

(Four Remarks)

Ω-invariance is weaker than Recursive

Ω-invariance is weaker than Recursive

Markov's principle MP can be reformulated as *If it is impossible that a TM runs forever, then it must halt*.

Ω-invariance is weaker than Recursive

Markov's principle MP can be reformulated as *If it is impossible that a TM runs forever, then it must halt*.

As no algorithmic upper bound on the halting time of the TM is given, MP is rejected in BISH.

Ω-invariance is weaker than Recursive

Markov's principle MP can be reformulated as *If it is impossible that a TM runs forever, then it must halt*.

As no algorithmic upper bound on the halting time of the TM is given, MP is rejected in BISH. The notion of algorithm in BISH is not identical to 'recursive'.

Ω-invariance is weaker than Recursive

Markov's principle MP can be reformulated as *If it is impossible that a TM runs forever, then it must halt*.

As no algorithmic upper bound on the halting time of the TM is given, MP is rejected in BISH. The notion of algorithm in BISH is not identical to 'recursive'.

Definition (In $\mathbb{NSA}$)

A formula ψ is Δ_1 if $\psi \iff (\exists n \in \mathbb{N})\varphi_1(n) \iff (\forall m \in \mathbb{N})\varphi_2(m)$.

Ω-invariance is weaker than Recursive

Markov's principle MP can be reformulated as *If it is impossible that a TM runs forever, then it must halt*.

As no algorithmic upper bound on the halting time of the TM is given, MP is rejected in BISH. The notion of algorithm in BISH is not identical to 'recursive'.

Definition (In $\mathbb{NSA}$)

A formula ψ is Δ_1 if $\psi \Longleftrightarrow (\exists n \in \mathbb{N})\varphi_1(n) \Longleftrightarrow (\forall m \in \mathbb{N})\varphi_2(m)$.

Theorem (In $\mathbb{NSA}$)

Only given $\mathbb{MP}$, every Δ_1-formula is decidable.

Ω-invariance is weaker than Recursive

Markov's principle MP can be reformulated as *If it is impossible that a TM runs forever, then it must halt*.

As no algorithmic upper bound on the halting time of the TM is given, MP is rejected in BISH. The notion of algorithm in BISH is not identical to 'recursive'.

Definition (In $\mathbb{NSA}$)

A formula ψ is Δ_1 if $\psi \iff (\exists n \in \mathbb{N})\varphi_1(n) \iff (\forall m \in \mathbb{N})\varphi_2(m)$.

Theorem (In $\mathbb{NSA}$)

Only given $\mathbb{MP}$, every Δ_1-formula is decidable.

But $\mathbb{MP}$ is not available in $\mathbb{NSA}$!

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

WKL (Every infinite tree $T \subset 2^{\mathbb{N}}$ has a path)

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

WKL (Every infinite tree $T \subset 2^{\mathbb{N}}$ has a path) is the classical contraposition of FAN_Δ and rejected in INT.

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

WKL (Every infinite tree $T \subset 2^{\mathbb{N}}$ has a path) is the classical contraposition of FAN_Δ and rejected in INT.

In BISH, we have $\mathsf{WKL} \to \mathsf{FAN}_\Delta$, and both are rejected.

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

WKL (Every infinite tree $T \subset 2^{\mathbb{N}}$ has a path) is the classical contraposition of FAN_Δ and rejected in INT.

In BISH, we have $\text{WKL} \rightarrow \text{FAN}_\Delta$, and both are rejected.

What happens in $\mathbb{NSA}$?

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

WKL (Every infinite tree $T \subset 2^{\mathbb{N}}$ has a path) is the classical contraposition of FAN_Δ and rejected in INT.

In BISH, we have WKL $\rightarrow \mathsf{FAN}_\Delta$, and both are rejected.

What happens in $\mathbb{NSA}$?

$\mathbb{WKL}$ $(\forall n \in \mathbb{N})(\exists \alpha \in 2^{\mathbb{N}})(\overline{\alpha}n \in T) \Rightarrow (\exists \alpha \in 2^{\mathbb{N}})(\forall n \in \mathbb{N})(\overline{\alpha}n \in T)$

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

WKL (Every infinite tree $T \subset 2^{\mathbb{N}}$ has a path) is the classical contraposition of FAN_Δ and rejected in INT.

In BISH, we have $\mathsf{WKL} \to \mathsf{FAN}_\Delta$, and both are rejected.

What happens in $\mathbb{NSA}$?

$\mathbb{WKL}(\forall n \in \mathbb{N})(\exists \alpha \in 2^{\mathbb{N}})(\overline{\alpha}n \in T) \Rightarrow (\exists \alpha \in 2^{\mathbb{N}})(\forall n \in \mathbb{N})(\overline{\alpha}n \in T)$

$\approx$ If the trees T and ${}^*T$ are (hyper)infinite, they share a path.

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

WKL (Every infinite tree $T \subset 2^{\mathbb{N}}$ has a path) is the classical contraposition of FAN_Δ and rejected in INT.

In BISH, we have WKL $\rightarrow$ FAN_Δ, and both are rejected.

What happens in $\mathbb{NSA}$?

$\mathbb{WKL}$ $(\forall n \in \mathbb{N})(\exists \alpha \in 2^{\mathbb{N}})(\overline{\alpha}n \in T) \Rightarrow (\exists \alpha \in 2^{\mathbb{N}})(\forall n \in \mathbb{N})(\overline{\alpha}n \in T)$

$\approx$ If the trees T and ${}^*T$ are (hyper)infinite, they share a path.

$\mathbb{FAN}_\Delta$

$(\forall \alpha \in 2^{\mathbb{N}})(\exists n \in \mathbb{N})(\overline{\alpha}n \in B) \Rightarrow (\exists k \in \mathbb{N})(\forall \alpha \in 2^{\mathbb{N}})(\exists n \leq k)(\overline{\alpha}n \in B)$

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

WKL (Every infinite tree $T \subset 2^{\mathbb{N}}$ has a path) is the classical contraposition of FAN_Δ and rejected in INT.

In BISH, we have WKL $\to$ FAN_Δ, and both are rejected.

What happens in $\mathbb{NSA}$?

$\mathbb{WKL}$ $(\forall n \in \mathbb{N})(\exists \alpha \in 2^{\mathbb{N}})(\overline{\alpha}n \in T) \Rightarrow (\exists \alpha \in 2^{\mathbb{N}})(\forall n \in \mathbb{N})(\overline{\alpha}n \in T)$

$\approx$ If the trees T and ${}^*T$ are (hyper)infinite, they share a path.

$\mathbb{FAN}_\Delta$
$(\forall \alpha \in 2^{\mathbb{N}})(\exists n \in \mathbb{N})(\overline{\alpha}n \in B) \Rightarrow (\exists k \in \mathbb{N})(\forall \alpha \in 2^{\mathbb{N}})(\exists n \leq k)(\overline{\alpha}n \in B)$

$\approx$ If a tree T is infinite, it has a path (${}^*T$ can be hyperfinite).

Fannying about: FAN_Δ vs WKL

FAN_Δ (Every detachable bar is uniform) is accepted in INT.

WKL (Every infinite tree $T \subset 2^{\mathbb{N}}$ has a path) is the classical contraposition of FAN_Δ and rejected in INT.

In BISH, we have $\mathsf{WKL} \rightarrow \mathsf{FAN}_\Delta$, and both are rejected.

What happens in $\mathbb{NSA}$?

$\mathbb{WKL}(\forall n \in \mathbb{N})(\exists \alpha \in 2^{\mathbb{N}})(\overline{\alpha} n \in T) \Rightarrow (\exists \alpha \in 2^{\mathbb{N}})(\forall n \in \mathbb{N})(\overline{\alpha} n \in T)$

$\approx$ If the trees T and ${}^*T$ are (hyper)infinite, they share a path.

$\mathbb{FAN}_\Delta$

$(\forall \alpha \in 2^{\mathbb{N}})(\exists n \in \mathbb{N})(\overline{\alpha} n \in B) \Rightarrow (\exists k \in \mathbb{N})(\forall \alpha \in 2^{\mathbb{N}})(\exists n \leq k)(\overline{\alpha} n \in B)$

$\approx$ If a tree T is infinite, it has a path (${}^*T$ can be hyperfinite).

In $\mathbb{NSA}$, we have $\mathbb{WKL} \Rightarrow \mathbb{FAN}_\Delta$.

A note on Coding and Assymetry

Recall that '$A \in \mathbb{T}$' means 'A satisfies Transfer'.

A note on Coding and Assymetry

Recall that '$A \in \mathbb{T}$' means 'A satisfies Transfer'.

E.g. '$(\forall n \in \mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\forall n \in \mathbb{N})\varphi(n) \to (\forall n \in {}^*\mathbb{N})\varphi(n)]$

A note on Coding and Assymetry

Recall that '$A \in \mathbb{T}$' means 'A satisfies Transfer'.

E.g. '$(\forall n \in \mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\forall n \in \mathbb{N})\varphi(n) \rightarrow (\forall n \in {}^*\mathbb{N})\varphi(n)]$
E.g. '$(\exists n \in {}^*\mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\exists n \in {}^*\mathbb{N})\varphi(n) \rightarrow (\exists n \in \mathbb{N}_1)\varphi(n)]$

A note on Coding and Assymetry

Recall that '$A \in \mathbb{T}$' means 'A satisfies Transfer'.

E.g. '$(\forall n \in \mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\forall n \in \mathbb{N})\varphi(n) \to (\forall n \in {}^*\mathbb{N})\varphi(n)]$
E.g. '$(\exists n \in {}^*\mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\exists n \in {}^*\mathbb{N})\varphi(n) \to (\exists n \in \mathbb{N}_1)\varphi(n)]$

Transfer is clearly asymmetric.

A note on Coding and Assymetry

Recall that '$A \in \mathbb{T}$' means 'A satisfies Transfer'.

E.g. '$(\forall n \in \mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\forall n \in \mathbb{N})\varphi(n) \rightarrow (\forall n \in {}^*\mathbb{N})\varphi(n)]$
E.g. '$(\exists n \in {}^*\mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\exists n \in {}^*\mathbb{N})\varphi(n) \rightarrow (\exists n \in \mathbb{N}_1)\varphi(n)]$

Transfer is clearly asymmetric.

First, to make hypernegation '$\sim$' work like intuitionistic negation.

A note on Coding and Assymetry

Recall that '$A \in \mathbb{T}$' means 'A satisfies Transfer'.

E.g. '$(\forall n \in \mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\forall n \in \mathbb{N})\varphi(n) \rightarrow (\forall n \in {}^*\mathbb{N})\varphi(n)]$
E.g. '$(\exists n \in {}^*\mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\exists n \in {}^*\mathbb{N})\varphi(n) \rightarrow (\exists n \in \mathbb{N}_1)\varphi(n)]$

Transfer is clearly asymmetric.

First, to make hypernegation '$\sim$' work like intuitionistic negation.

Secondly, for fundamental reasons:

A note on Coding and Assymetry

Recall that '$A \in \mathbb{T}$' means 'A satisfies Transfer'.

E.g. '$(\forall n \in \mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\forall n \in \mathbb{N})\varphi(n) \to (\forall n \in {}^*\mathbb{N})\varphi(n)]$
E.g. '$(\exists n \in {}^*\mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\exists n \in {}^*\mathbb{N})\varphi(n) \to (\exists n \in \mathbb{N}_1)\varphi(n)]$

Transfer is clearly asymmetric.

First, to make hypernegation '$\sim$' work like intuitionistic negation.

Secondly, for fundamental reasons:

In '$(\exists n_0 \in {}^*\mathbb{N})\varphi(n_0)$', the number n_0 could be a code for some $f : \mathbb{N} \to \mathbb{N}$ (Keisler).

A note on Coding and Assymetry

Recall that '$A \in \mathbb{T}$' means 'A satisfies Transfer'.

E.g. '$(\forall n \in \mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\forall n \in \mathbb{N})\varphi(n) \rightarrow (\forall n \in {}^*\mathbb{N})\varphi(n)]$
E.g. '$(\exists n \in {}^*\mathbb{N})\varphi(n) \in \mathbb{T}$' is $[(\exists n \in {}^*\mathbb{N})\varphi(n) \rightarrow (\exists n \in \mathbb{N}_1)\varphi(n)]$

Transfer is clearly asymmetric.

First, to make hypernegation '$\sim$' work like intuitionistic negation.

Secondly, for fundamental reasons:

In '$(\exists n_0 \in {}^*\mathbb{N})\varphi(n_0)$', the number n_0 could be a code for some $f : \mathbb{N} \rightarrow \mathbb{N}$ (Keisler). If '$(\exists n_0 \in {}^*\mathbb{N})\varphi(n_0)$' implies '$(\exists n_1 \in \mathbb{N})\varphi(n_1)$', then f has a finite code $n_1 \in \mathbb{N}$, making its graph Δ_0.

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for 'mixed' theorems:

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for 'mixed' theorems:

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for 'mixed' theorems:

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
LPO $\leftrightarrow$ MP+WLPO	

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for 'mixed' theorems:

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
LPO $\leftrightarrow$ MP+WLPO MP $\leftrightarrow$ WMP + $\text{MP}^\vee$	

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for 'mixed' theorems:

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
LPO $\leftrightarrow$ MP+WLPO MP $\leftrightarrow$ WMP + $\text{MP}^\vee$ WLPO $\rightarrow$ LLPO	

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for ‘mixed’ theorems:

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
LPO $\leftrightarrow$ MP+WLPO	
MP $\leftrightarrow$ WMP + $\text{MP}^\vee$	
WLPO $\to$ LLPO	
LLPO $\to$ $\text{MP}^\vee$	

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for 'mixed' theorems:

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
LPO $\leftrightarrow$ MP+WLPO MP $\leftrightarrow$ WMP + $\text{MP}^\vee$ WLPO $\rightarrow$ LLPO LLPO $\rightarrow$ $\text{MP}^\vee$ LPO $\rightarrow$ BD-N	

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for 'mixed' theorems:

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
LPO $\leftrightarrow$ MP+WLPO	
MP $\leftrightarrow$ WMP + $\text{MP}^\vee$	
WLPO $\rightarrow$ LLPO	
LLPO $\rightarrow$ $\text{MP}^\vee$	
LPO $\rightarrow$ BD-N	
LLPO $\rightarrow$ FAN_Δ	

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for 'mixed' theorems:

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
LPO $\leftrightarrow$ MP+WLPO MP $\leftrightarrow$ WMP + $\text{MP}^\vee$ WLPO $\rightarrow$ LLPO LLPO $\rightarrow$ $\text{MP}^\vee$ LPO $\rightarrow$ BD-N LLPO $\rightarrow$ FAN_Δ LLPO $\leftrightarrow$ WKL	

Constructive Reverse Mathematics under $\mathbb{B}$ IV

Same for WMP, FAN_Δ, BD-N, and $\text{MP}^\vee$.
Same for 'mixed' theorems:

BISH (based on BHK)	$\mathbb{NSA}$ (based on CL)
$\text{LPO} \leftrightarrow \text{MP}+\text{WLPO}$	$\mathbb{LPO} \Longleftrightarrow \mathbb{MP} + \mathbb{WLPO}$
$\text{MP} \leftrightarrow \text{WMP} + \text{MP}^\vee$	$\mathbb{MP} \Longleftrightarrow \mathbb{WMP} + \mathbb{MP}^\vee$
$\text{WLPO} \rightarrow \text{LLPO}$	$\mathbb{WLPO} \Rrightarrow \mathbb{LLPO}$
$\text{LLPO} \rightarrow \text{MP}^\vee$	$\mathbb{LLPO} \Rrightarrow \mathbb{MP}^\vee$
$\text{LPO} \rightarrow \text{BD-N}$	$\mathbb{LPO} \Rrightarrow \mathbb{BD}\text{-}\mathbb{N}$
$\text{LLPO} \rightarrow \text{FAN}_\Delta$	$\mathbb{LLPO} \Rrightarrow \mathbb{FAN}_\Delta$
$\text{LLPO} \leftrightarrow \text{WKL}$	$\mathbb{LLPO} \Longleftrightarrow \mathbb{WKL}$

Conclusion: $\mathbb{NSA} \approx$ BISH

Conclusion: $\mathbb{NSA} \approx$ BISH

If BISH $\vdash X$ then $X \not\to$ LPO, LLPO, MP, ... (princ. rejected in BISH)

If $\mathbb{NSA} \vdash Y$ then $Y \not\Rightarrow \mathbb{LPO}, \mathbb{LLPO}, \mathbb{MP}, \ldots$ (not provable in $\mathbb{NSA}$)

Conclusion: $\mathbb{NSA} \approx$ BISH

If BISH $\vdash X$ then $X \not\to$ LPO, LLPO, MP, ... (princ. rejected in BISH)

If $\mathbb{NSA} \vdash Y$ then $Y \not\Rightarrow \mathbb{LPO}, \mathbb{LLPO}, \mathbb{MP}, \dots$ (not provable in $\mathbb{NSA}$)

Reuniting the antipodes (Palmgren & Moerdijk).

Conclusion: $\mathbb{NSA} \approx$ BISH

If BISH $\vdash X$ then $X \not\to$ LPO, LLPO, MP, ... (princ. rejected in BISH)

If $\mathbb{NSA} \vdash Y$ then $Y \not\Rightarrow \mathbb{LPO}, \mathbb{LLPO}, \mathbb{MP}, \dots$ (not provable in $\mathbb{NSA}$)

Reuniting the antipodes (Palmgren & Moerdijk).

Reverse-engineering Reverse Mathematics (Fuchino-sensei)

Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

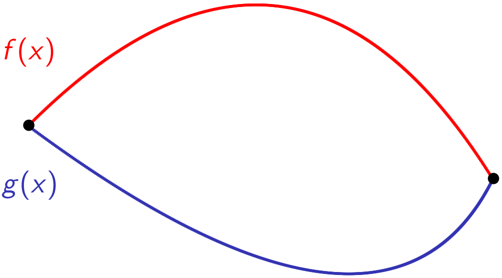
Can Ω-invariance help capture e.g. Type Theory?

Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Can Ω-invariance help capture e.g. Type Theory?

Homotopy:

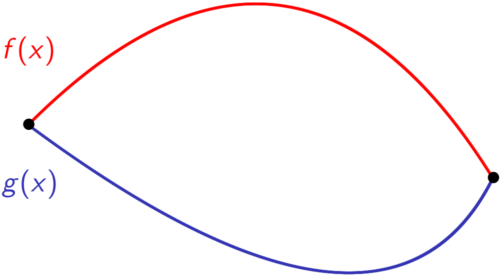


Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Can Ω-invariance help capture e.g. Type Theory?

Homotopy: continuous transformation h_t of f to g ($t \in [0,1]$).

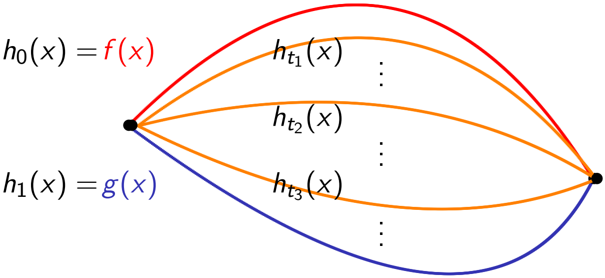


Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Can Ω-invariance help capture e.g. Type Theory?

Homotopy: continuous transformation h_t of f to g ($t \in [0,1]$).

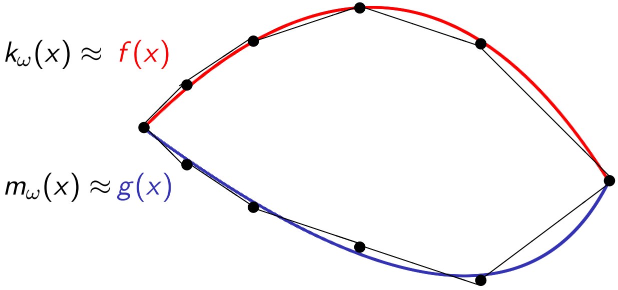


Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Can Ω-invariance help capture e.g. Type Theory?

Homotopy:

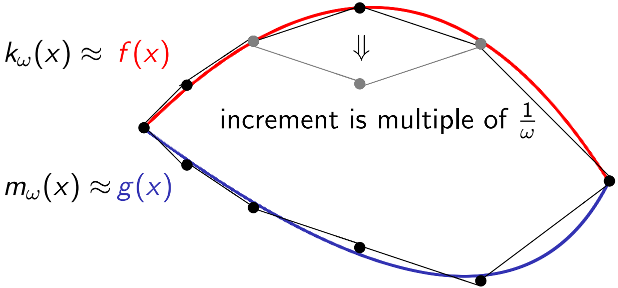


Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Can Ω-invariance help capture e.g. Type Theory?

Homotopy:

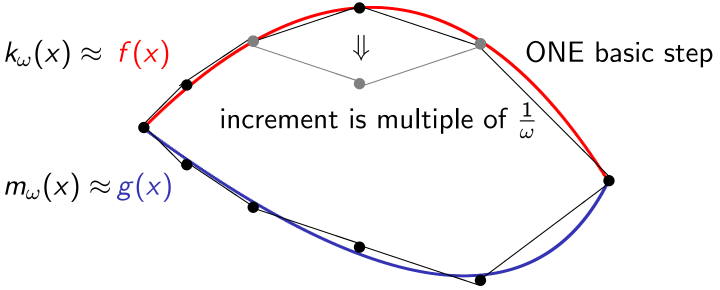


Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Can Ω-invariance help capture e.g. Type Theory?

Homotopy:

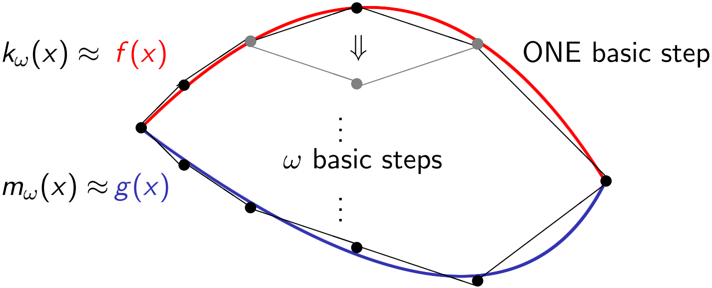


Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Can Ω-invariance help capture e.g. Type Theory?

Homotopy:

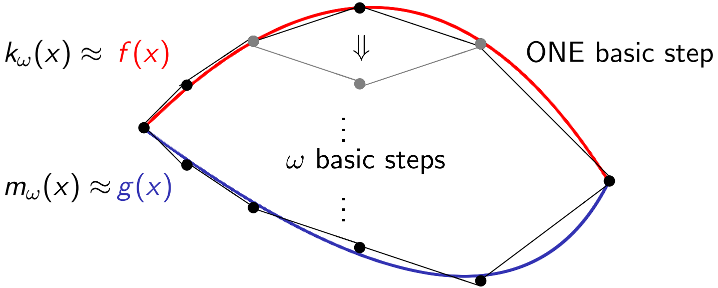


Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Can Ω-invariance help capture e.g. Type Theory?

Homotopy:



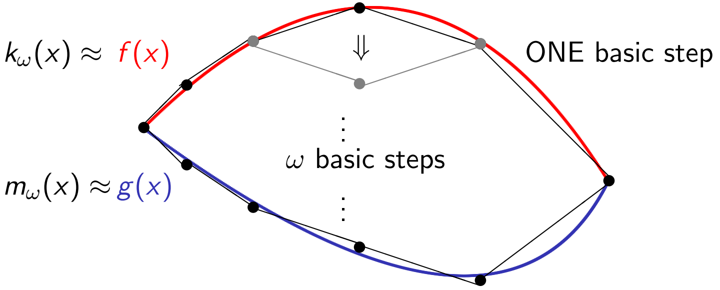
Independent of the choice of ω

Future work: Type Theory

Martin-Löf intended his type theory as a foundation for BISH.

Can Ω-invariance help capture e.g. Type Theory?

Homotopy: $\approx$ Ω-invariant broken-line transformation $h_{\omega,t}$ of f to g.



Independent of the choice of ω

Philosophy of Physics

Philosophy of Physics

Why is Mathematics in Physics so constructive/computable?

Philosophy of Physics

Why is Mathematics in Physics so constructive/computable?

Indeed, most of Physics can be formalized in BISH (e.g. Gleason's theorem).

Philosophy of Physics

Why is Mathematics in Physics so constructive/computable?

Indeed, most of Physics can be formalized in BISH (e.g. Gleason's theorem).

Yet, in Physics, an informal version of NSA is used to date.

Philosophy of Physics

Why is Mathematics in Physics so constructive/computable?

Indeed, most of Physics can be formalized in BISH (e.g. Gleason's theorem).

Yet, in Physics, an informal version of NSA is used to date. (Weierstraß' notorious 'ε-δ' method was never adopted, neither was BISH).

Philosophy of Physics

Why is Mathematics in Physics so constructive/computable?

Indeed, most of Physics can be formalized in BISH (e.g. Gleason's theorem).

Yet, in Physics, an informal version of NSA is used to date. (Weierstraß' notorious 'ε-δ' method was never adopted, neither was BISH).

Now, in Physics, the end result of a calculation should have physical meaning (modeling of reality).

Philosophy of Physics

Why is Mathematics in Physics so constructive/computable?

Indeed, most of Physics can be formalized in BISH (e.g. Gleason's theorem).

Yet, in Physics, an informal version of NSA is used to date. (Weierstraß' notorious 'ε-δ' method was never adopted, neither was BISH).

Now, in Physics, the end result of a calculation should have physical meaning (modeling of reality).

A mathematical result with physical meaning will not depend on the choice of infinite number/infinitesimal used, i.e. it is Ω-invariant. (Alain Connes)

Philosophy of Mathematics: Whither Structuralism?

Philosophy of Mathematics: Whither Structuralism?

Structuralism ≈ Mathematics is about a single structure.

Philosophy of Mathematics: Whither Structuralism?

Structuralism $\approx$ Mathematics is about a single structure.

E.g. first-order arithmetic is about (models isomorphic to) the standard model $\mathbb{N}$.

Philosophy of Mathematics: Whither Structuralism?

Structuralism $\approx$ Mathematics is about a single structure.

E.g. first-order arithmetic is about (models isomorphic to) the standard model $\mathbb{N}$.

Problem: How to exclude the nonstandard models of arithmetic? (Second-order?, Tennenbaum's Theorem?)

Philosophy of Mathematics: Whither Structuralism?

Structuralism $\approx$ Mathematics is about a single structure.

E.g. first-order arithmetic is about (models isomorphic to) the standard model $\mathbb{N}$.

Problem: How to exclude the nonstandard models of arithmetic? (Second-order?, Tennenbaum's Theorem?)

When life gives you lemons... you make Ω-invariance:

Philosophy of Mathematics: Whither Structuralism?

Structuralism $\approx$ Mathematics is about a single structure.

E.g. first-order arithmetic is about (models isomorphic to) the standard model $\mathbb{N}$.

Problem: How to exclude the nonstandard models of arithmetic? (Second-order?, Tennenbaum's Theorem?)

When life gives you lemons... you make Ω-invariance:

Arithmetic is about a computationally robust variety of structures.

Philosophy of Mathematics: Whither Structuralism?

Structuralism $\approx$ Mathematics is about a single structure.

E.g. first-order arithmetic is about (models isomorphic to) the standard model $\mathbb{N}$.

Problem: How to exclude the nonstandard models of arithmetic? (Second-order?, Tennenbaum's Theorem?)

When life gives you lemons... you make Ω-invariance:

Arithmetic is about a computationally robust variety of structures.

Despite Tennenbaum's Theorem, one can define computability/constructivity via Ω-invariance in each nonstandard model of arithmetic.

Final Thoughts

Final Thoughts

And what are these [infinitesimals]? [...] They are neither finite Quantities nor Quantities infinitely small, nor yet nothing. May we not call them the ghosts of departed quantities?

George Berkeley, *The Analyst*

Final Thoughts

And what are these [infinitesimals]? [...] They are neither finite Quantities nor Quantities infinitely small, nor yet nothing. May we not call them the ghosts of departed quantities?
George Berkeley, *The Analyst*

...there are good reasons to believe that Nonstandard Analysis, in some version or other, will be the analysis of the future.
Kurt Gödel

Final Thoughts

And what are these [infinitesimals]? [. . .] They are neither finite Quantities nor Quantities infinitely small, nor yet nothing. May we not call them the ghosts of departed quantities?
George Berkeley, *The Analyst*

...there are good reasons to believe that Nonstandard Analysis, in some version or other, will be the analysis of the future.
Kurt Gödel

We thank the John Templeton Foundation for its generous support!

Final Thoughts

And what are these [infinitesimals]? [...] They are neither finite Quantities nor Quantities infinitely small, nor yet nothing. May we not call them the ghosts of departed quantities?
George Berkeley, *The Analyst*

...there are good reasons to believe that Nonstandard Analysis, in some version or other, will be the analysis of the future.
Kurt Gödel

We thank the John Templeton Foundation for its generous support!

Thank you for your attention!

Final Thoughts

And what are these [infinitesimals]? [...] They are neither finite Quantities nor Quantities infinitely small, nor yet nothing. May we not call them the ghosts of departed quantities?
George Berkeley, *The Analyst*

...there are good reasons to believe that Nonstandard Analysis, in some version or other, will be the analysis of the future.
Kurt Gödel

We thank the John Templeton Foundation for its generous support!

Thank you for your attention!

Any questions?

Take-home message

In Nonstandard Analysis, an algorithm is any object whose definition is independent of the choice of infinitesimal (Ω-invariance).

More technically, we define a translation between Constructive Analysis (BISH) and Nonstandard Analysis (NSA):

(Proof and Algorithm) in BISH = (Transfer and Ω-invariance) in NSA

Most results from CRM (= RM based on BISH) translate to NSA via a natural translation $\mathbb{B}$.