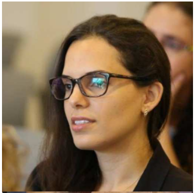
Open Bar: a Brouwerian Intuitionistic Logic with a Pinch of Excluded Middle

Mark Bickford, Liron Cohen, Bob Constable, Vincent Rahli

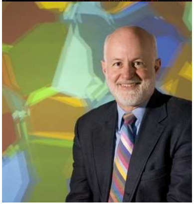
January 27, 2021

In collaboration with



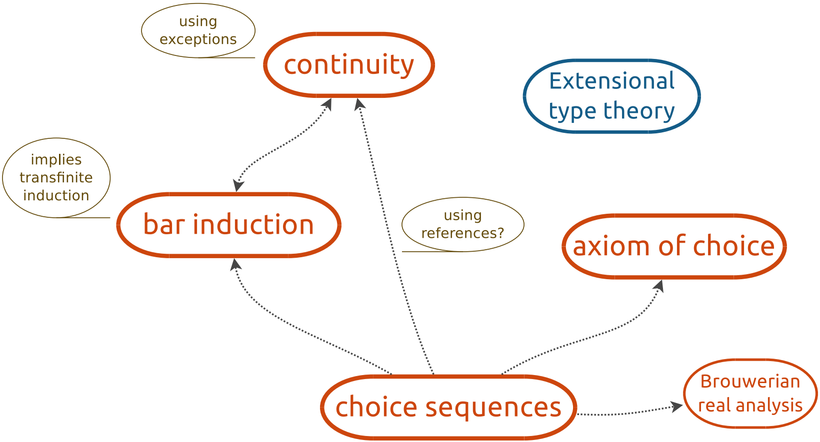
Liron Cohen

Mark Bickford



Bob Constable

Towards a Brouwerian Intuitionistic Logic



Towards a Brouwerian Intuitionistic Logic

1952: Kleene proved that the Baire space ($\mathbb{N}^{\mathbb{N}}$) cannot be restricted to general recursive functions for the Fan Theorem and Bar Induction to hold.

Towards a Brouwerian Intuitionistic Logic

1952: Kleene proved that the Baire space ($\mathbb{N}^{\mathbb{N}}$) cannot be restricted to general recursive functions for the Fan Theorem and Bar Induction to hold.

2017: We initially added infinite sequences to the Baire space

Towards a Brouwerian Intuitionistic Logic

1952: Kleene proved that the Baire space ($\mathbb{N}^{\mathbb{N}}$) cannot be restricted to general recursive functions for the Fan Theorem and Bar Induction to hold.

2017: We initially added infinite sequences to the Baire space

Due to our realizability interpretation, some properties became undecidable, e.g., decidability of syntactic equality

Towards a Brouwerian Intuitionistic Logic

1952: Kleene proved that the Baire space ($\mathbb{N}^{\mathbb{N}}$) cannot be restricted to general recursive functions for the Fan Theorem and Bar Induction to hold.

2017: We initially added infinite sequences to the Baire space

Due to our realizability interpretation, some properties became undecidable, e.g., decidability of syntactic equality

2018: (BITT) Replaced infinite sequences by choice sequences

- Contradict classical axioms!
- Enough to validate Bar Induction?

Towards a Brouwerian Intuitionistic Logic

1952: Kleene proved that the Baire space ($\mathbb{N}^{\mathbb{N}}$) cannot be restricted to general recursive functions for the Fan Theorem and Bar Induction to hold.

2017: We initially added infinite sequences to the Baire space

Due to our realizability interpretation, some properties became undecidable, e.g., decidability of syntactic equality

2018: (BITT) Replaced infinite sequences by choice sequences

- Contradict classical axioms!
- Enough to validate Bar Induction?

2020: (OpenTT) Relaxed model to validate classical axioms

- Consistent with classical axioms!
- Enough to validate Bar Induction?

Starting point: an Extensional Type Theory

Untyped call-by-name lambda-calculus

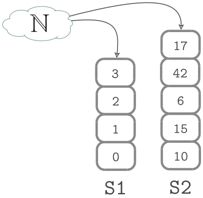
Introduction/elimination deduction rules

realizability semantics

- Extensional
- Dependent types
- Partial types

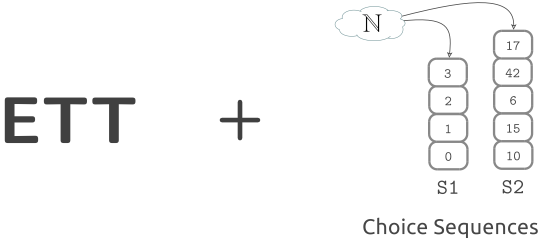
Adding Choice Sequences

ETT +



Choice Sequences

Adding Choice Sequences



Broader sense of computation

Adding Choice Sequences

lawless (free choice) sequences: no restrictions on the choices (except for initial segments)

LS$_1$ (density) $\forall s.\exists\alpha.\alpha \in s$

LS$_2$ (discreteness) $\forall\alpha,\beta.(\alpha \equiv \beta \ \vee \ \neg\alpha \equiv \beta)$

LS$_3$ (open data) $A(\alpha) \ \Rightarrow \ \exists n.\forall\beta.(\overline{\alpha}n = \overline{\beta}n \ \Rightarrow \ A(\beta))$

s	finite sequence
α	lawless sequence
$\alpha \in s$	s is an initial segment of α
$\equiv$	intensional equality
$\overline{\alpha}n$	the initial segment of α of length n

Adding Choice Sequences

BITT

LS1 $\Pi n{:}\mathbb{N}.\Pi f{:}\mathcal{B}_n.\Sigma\alpha{:}\texttt{Free}.f = \alpha \in \mathcal{B}_n$

LS2 $\Pi\alpha, \beta{:}\texttt{Free}.(\alpha{=}\beta \in \mathcal{B}){+}(\neg\alpha{=}\beta \in \mathcal{B})$

LS3 –

¬LEM $\neg\Pi P{:}\mathbb{P}.{\downarrow}(P{+}\neg P)$

¬MP $\neg\Pi P{:}\mathbb{B}^{\mathbb{N}}.\neg(\Pi n{:}\mathbb{N}.\neg P(n)) \rightarrow \Sigma n{:}\mathbb{N}.P(n)$

¬IP $\neg\Pi A{:}\mathbb{P}.\Pi B{:}\mathbb{P}^{\mathbb{N}}.(A \rightarrow \Sigma n{:}\mathbb{N}.B(n)) \rightarrow \Sigma n{:}\mathbb{N}.(A \rightarrow B(n))$

¬LPO $\neg\Pi P{:}\mathbb{B}^{\mathbb{N}}.(\Sigma\mathbb{N}{:}n.P(n)){+}(\Pi n{:}\mathbb{N}.\neg P(n))$

OpenTT

LS1 $\Pi n{:}\mathbb{N}.\Pi f{:}\mathcal{B}_n.{\downarrow}\Sigma\alpha{:}\texttt{Free}.f = \alpha \in \mathcal{B}_n$

LS2 $\Pi\alpha, \beta{:}\texttt{Free}.(\alpha{=}\beta \in \mathcal{B}){+}(\neg\alpha{=}\beta \in \mathcal{B})$

LS3 $\Pi\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \Sigma n{:}\mathbb{N}_{\wr}.\Pi\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_{\wr n} \rightarrow {\downarrow}P(\beta))$

LEM $\Pi P{:}\mathbb{P}.{\downarrow}(P{+}\neg P)$

(where $\mathcal{B} = \mathbb{N}^{\mathbb{N}}$ and $\mathcal{B}_n = \mathbb{N}^{\mathbb{N}_n}$)
($\downarrow$ is a "proof erasure" operator)

Syntax & Operational Semantics

Syntax:

$$
\begin{aligned}
T \in \texttt{Type} ::= {} & \mathbb{N} \mid \mathbb{U}_i \mid \mathbf{\Pi} x{:}t.t \mid \mathbf{\Sigma} x{:}t.t \mid \{x : t \mid t\} \\
\mid {} & t = t \in t \mid t{+}t \mid \ldots \\
\mid {} & \texttt{Free} \text{ (choice sequence type)}
\end{aligned}
$$

$$
\begin{aligned}
v \in \texttt{Value} ::= {} & T \mid \star \mid \underline{n} \mid \lambda x.t \mid \langle t, t \rangle \mid \texttt{inl}(t) \mid \texttt{inr}(t) \mid \ldots \\
\mid {} & \eta \text{ (choice sequence name)}
\end{aligned}
$$

$$
\begin{aligned}
t \in \texttt{Term} ::= {} & x \mid v \mid t\ t \mid \texttt{fix}(t) \mid \texttt{let } x := t \texttt{ in } t \\
\mid {} & \texttt{case } t \texttt{ of inl}(x) \Rightarrow t \texttt{ | inr}(y) \Rightarrow t \\
\mid {} & \texttt{let } x, y = t \texttt{ in } t \mid \texttt{if } t{=}t \texttt{ then } t \texttt{ else } t \mid \ldots
\end{aligned}
$$

Syntax & Operational Semantics

Syntax:

$$
\begin{array}{lll}
T \in \texttt{Type} & ::= & \mathbb{N} \mid \mathbb{U}_i \mid \mathbf{\Pi} x{:}t.t \mid \mathbf{\Sigma} x{:}t.t \mid \{x : t \mid t\} \\
 & \mid & t = t \in t \mid t{+}t \mid \dots \\
 & \mid & \texttt{Free} \text{ (choice sequence type)}
\end{array}
$$

$$
\begin{array}{lll}
v \in \texttt{Value} & ::= & T \mid \star \mid \underline{n} \mid \lambda x.t \mid \langle t, t \rangle \mid \texttt{inl}(t) \mid \texttt{inr}(t) \mid \dots \\
 & \mid & \eta \text{ (choice sequence name)}
\end{array}
$$

$$
\begin{array}{lll}
t \in \texttt{Term} & ::= & x \mid v \mid t\ t \mid \texttt{fix}(t) \mid \texttt{let}\ x := t\ \texttt{in}\ t \\
 & \mid & \texttt{case}\ t\ \texttt{of}\ \texttt{inl}(x) \Rightarrow t \mid \texttt{inr}(y) \Rightarrow t \\
 & \mid & \texttt{let}\ x, y = t\ \texttt{in}\ t \mid \texttt{if}\ t{=}t\ \texttt{then}\ t\ \texttt{else}\ t \mid \dots
\end{array}
$$

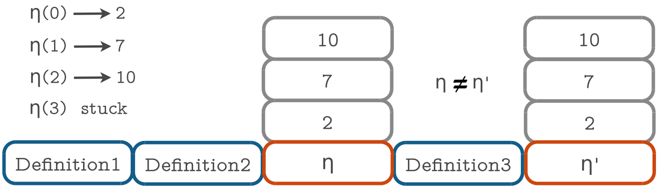
Operational semantics:

$$
\begin{array}{lcl}
(\lambda x.t_1)\ t_2 & \longmapsto & t_1[x \backslash t_2] \\
\texttt{let}\ x_1, x_2 = \langle t_1, t_2 \rangle\ \texttt{in}\ t & \longmapsto & t[x_1 \backslash t_1; x_2 \backslash t_2] \\
\texttt{fix}(v) & \longmapsto & v\ \texttt{fix}(v)
\end{array}
$$

...

World-Based Computations

World-dependent operational semantics:

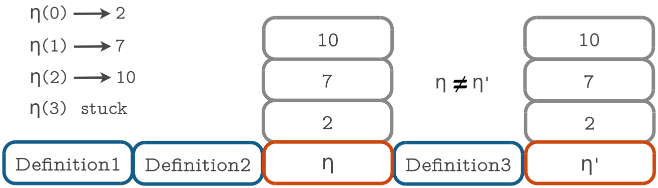


Worlds include:

- definitions
- choice sequences

World-Based Computations

World-dependent operational semantics:



Worlds include:

- definitions
- choice sequences

Worlds can be extended

- horizontally
- vertically
- $w_2 \succeq w_1$

OpenTT

Standard ETT rules:

$$\frac{\Gamma, x : A \vdash b : B[x] \qquad \Gamma \vdash \star : (A \in \mathbb{U}_i)}{\Gamma \vdash \lambda x.b : \mathbf{\Pi} a{:}A.B[a]} \qquad \cdots$$

OpenTT

Standard ETT rules:

$$\frac{\Gamma, x : A \vdash b : B[x] \qquad \Gamma \vdash \star : (A \in \mathbb{U}_i)}{\Gamma \vdash \lambda x.b : \mathbf{\Pi} a{:}A.B[a]} \qquad \cdots$$

+ choice sequence rules:

$$\frac{}{\Gamma \vdash \star : (\eta \in \mathtt{Free})} \qquad \frac{}{\Gamma \vdash \star : (\eta \in \mathcal{B})} \qquad \cdots$$

OpenTT

Standard ETT rules:

$$\frac{\Gamma, x : A \vdash b : B[x] \qquad \Gamma \vdash \star : (A \in \mathbb{U}_i)}{\Gamma \vdash \lambda x.b : \mathbf{\Pi} a{:}A.B[a]} \qquad \cdots$$

+ choice sequence rules:

$$\frac{}{\Gamma \vdash \star : (\eta \in \texttt{Free})} \qquad \frac{}{\Gamma \vdash \star : (\eta \in \mathcal{B})} \qquad \cdots$$

+ LS1 (density), LS2 (discreteness), LS3 (Open Data)

OpenTT

Standard ETT rules:

$$\frac{\Gamma, x : A \vdash b : B[x] \qquad \Gamma \vdash \star : (A \in \mathbb{U}_i)}{\Gamma \vdash \lambda x.b : \mathbf{\Pi} a{:}A.B[a]} \qquad \cdots$$

+ choice sequence rules:

$$\frac{}{\Gamma \vdash \star : (\eta \in \texttt{Free})} \qquad \frac{}{\Gamma \vdash \star : (\eta \in \mathcal{B})} \qquad \cdots$$

+ LS1 (density), LS2 (discreteness), LS3 (Open Data)

+ LEM

OpenTT

Standard ETT rules:

$$\frac{\Gamma, x : A \vdash b : B[x] \qquad \Gamma \vdash \star : (A \in \mathbb{U}_i)}{\Gamma \vdash \lambda x.b : \mathbf{\Pi} a{:}A.B[a]} \qquad \cdots$$

+ choice sequence rules:

$$\frac{}{\Gamma \vdash \star : (\eta \in \mathtt{Free})} \qquad \frac{}{\Gamma \vdash \star : (\eta \in \mathcal{B})} \qquad \cdots$$

+ LS1 (density), LS2 (discreteness), LS3 (Open Data)

+ LEM

- How do we validate these rules?

OpenTT

Standard ETT rules:

$$\frac{\Gamma, x : A \vdash b : B[x] \qquad \Gamma \vdash \star : (A \in \mathbb{U}_i)}{\Gamma \vdash \lambda x.b : \mathbf{\Pi} a{:}A.B[a]} \qquad \cdots$$

+ **choice sequence rules**:

$$\frac{}{\Gamma \vdash \star : (\eta \in \texttt{Free})} \qquad \frac{}{\Gamma \vdash \star : (\eta \in \mathcal{B})} \qquad \cdots$$

+ **LS1 (density), LS2 (discreteness), LS3 (Open Data)**

+ **LEM**

- How do we validate these rules?
- Why is LEM an OpenTT rule, but not a BITT rule?

Realizability semantics

An inductive relation that expresses type equality

$T_1 \equiv T_2$ $\qquad\qquad$ $\texttt{type}(T)$ is $T \equiv T$

A recursive function that expresses equality in a type

$$a \equiv b \in T$$

Realizability semantics

An inductive relation that expresses type equality

$$T_1 \equiv T_2 \qquad\qquad \texttt{type}(T) \text{ is } T \equiv T$$

A recursive function that expresses equality in a type

$$a \equiv b \in T$$

For example (product types):

$f_1 \equiv f_2 \in \mathbf{\Pi} x{:}A.B$ $\qquad$ $\texttt{type}((\mathbf{\Pi} x{:}A.B)) \land$
$\forall a_1, a_2.\ a_1 \equiv a_2 \in A \Rightarrow f_1(a_1) \equiv f_2(a_2) \in B[x \backslash a_1]$

$\mathbf{\Pi} x_1{:}A_1.B_1 \equiv \mathbf{\Pi} x_2{:}A_2.B_2$ $\qquad$ $A_1 \equiv A_2 \land$
$\forall a_1, a_2.\ a_1 \equiv a_2 \in A_1 \Rightarrow B_1[x_1 \backslash a_1] \equiv B_2[x_2 \backslash a_2]$

Realizability semantics

An inductive relation that expresses type equality

$T_1 \equiv T_2$ $\qquad$ $\texttt{type}(T)$ is $T \equiv T$

A recursive function that expresses equality in a type

$$a \equiv b \in T$$

For example (product types):

$f_1 \equiv f_2 \in \mathbf{\Pi} x{:}A.B$	$\texttt{type}((\mathbf{\Pi} x{:}A.B)) \wedge$ $\forall a_1, a_2.\ a_1 \equiv a_2 \in A \Rightarrow f_1(a_1) \equiv f_2(a_2) \in B[x \backslash a_1]$
$\mathbf{\Pi} x_1{:}A_1.B_1 \equiv \mathbf{\Pi} x_2{:}A_2.B_2$	$A_1 \equiv A_2 \wedge$ $\forall a_1, a_2.\ a_1 \equiv a_2 \in A_1 \Rightarrow B_1[x_1 \backslash a_1] \equiv B_2[x_2 \backslash a_2]$

Enough to validate choice sequence rules?

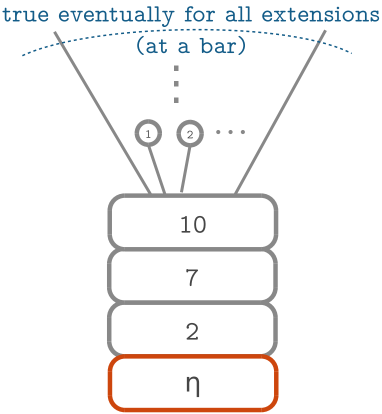
Going back to BITT for a minute

Why is $\eta \in \mathcal{B}$ valid in BITT?

Going back to BITT for a minute

Why is $\eta \in \mathcal{B}$ valid in BITT?

We used a Beth interpretation:



Formally:

$\eta \in \mathcal{B}$ **is true in world** w

$\iff$

$\forall m : \textbf{nat}.\exists b : \text{bar}(w).\forall w' \in b.$
$\eta(m)$ **computes to a nat in** w'

Going back to BITT for a minute

This model rules out a number of axioms (e.g., LEM)

Going back to BITT for a minute

This model rules out a number of axioms (e.g., LEM)

Why?

Going back to BITT for a minute

This model rules out a number of axioms (e.g., LEM)

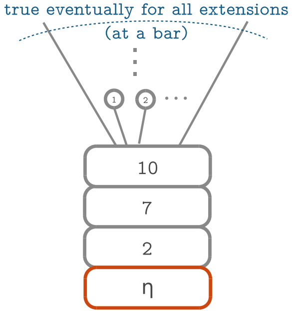
Why?

Given a "fresh" (no choices so far) free choice sequence α,

- it is not true that $\boldsymbol{\Sigma} n{:}\mathbb{N}.\alpha(n) = 1 \in \mathbb{N}$ because there is a path where α is only extended with 0
- it is not true that $\neg\boldsymbol{\Sigma} n{:}\mathbb{N}.\alpha(n) = 1 \in \mathbb{N}$ because there is path where α is extended with 1
  - The meaning of $\neg T$ is that T is false in all extensions

Any way around this?

Beth model

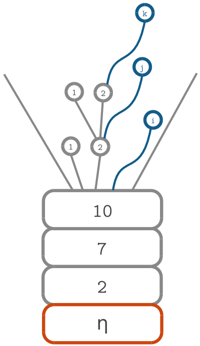


$T@w$

$\iff$

$\exists b \in \mathsf{bar}(w).\forall w_1 \in b.$
$\forall w_2 \succeq w_1.T@w_2$

Open Bar model



$T@w$

$\iff$

$\forall w_1 \succeq w.\exists w_2 \succeq w_1.$
$\forall w_3 \succeq w_2.T@w_3$

Open Bar model

$$T@w \iff \forall w_1 \succeq w.\exists w_2 \succeq w_1.\forall w_3 \succeq w_2.T@w_3$$

Open Bar model

$$T@w \iff \forall w_1 \succeq w.\exists w_2 \succeq w_1.\forall w_3 \succeq w_2.T@w_3$$

Bears a resemblance to double negation translation:

- Kripke interpretation of $A \rightarrow B$:

$$[\![A \rightarrow B]\!]_w = \forall w_1 \succeq w.[\![A]\!]_{w_1} \Rightarrow [\![B]\!]_{w_1}$$

Open Bar model

$$T@w \iff \forall w_1 \succeq w.\exists w_2 \succeq w_1.\forall w_3 \succeq w_2.T@w_3$$

Bears a resemblance to double negation translation:

- Kripke interpretation of $A \rightarrow B$:

$$[\![A \rightarrow B]\!]_w = \forall w_1 \succeq w.[\![A]\!]_{w_1} \Rightarrow [\![B]\!]_{w_1}$$

- In such a semantics, $\neg\neg A$ is interpreted as:

$$\forall w_1 \succeq w.\neg\forall w_2 \succeq w_1.\neg[\![A]\!]_{w_2}$$

Open Bar model

$$T@w \iff \forall w_1 \succeq w.\exists w_2 \succeq w_1.\forall w_3 \succeq w_2.T@w_3$$

Bears a resemblance to double negation translation:

- Kripke interpretation of $A \to B$:

$$[\![A \to B]\!]_w = \forall w_1 \succeq w.[\![A]\!]_{w_1} \Rightarrow [\![B]\!]_{w_1}$$

- In such a semantics, $\neg\neg A$ is interpreted as:

$$\forall w_1 \succeq w.\neg\forall w_2 \succeq w_1.\neg[\![A]\!]_{w_2}$$

- classically equivalent to:

$$\forall w_1 \succeq w.\exists w_2 \succeq w_1.[\![A]\!]_{w_2}$$

LEM in the Open Bar model

This model still satisfies the choice sequence axioms

LEM in the Open Bar model

This model still satisfies the choice sequence axioms

LEM can now be validated using classical reasoning:

$$\frac{}{\Gamma \vdash \lambda P.\star : \mathbf{\Pi} P{:}\mathbb{P}.{\downarrow}(P + \neg P)}$$

LEM in the Open Bar model

This model still satisfies the choice sequence axioms

LEM can now be validated using classical reasoning:

$$\frac{}{\Gamma \vdash \lambda P.\star : \mathbf{\Pi} P{:}\mathbb{P}.{\downarrow}(P + \neg P)}$$

- Let w_0 be the current world
- $\forall w_1 \succeq w_0$, using classical reasoning we can assume that
  - either $\exists w_2 \succeq w_1.P@w_2$
  - or $\neg\exists w_2 \succeq w_1.P@w_2$

  Either way we conclude trivially

Choice sequences in the Open Bar model

LS1 (density) is valid

$$\frac{}{\Gamma \vdash \lambda n, f.\star : \mathbf{\Pi} n{:}\mathbb{N}.\mathbf{\Pi} f{:}\mathcal{B}_n.\downarrow\mathbf{\Sigma}\alpha{:}\texttt{Free}.f = \alpha \in \mathcal{B}_n}$$

Choice sequences in the Open Bar model

LS1 (density) is valid

$$\overline{\Gamma \vdash \lambda n, f.\star : \Pi n{:}\mathbb{N}.\Pi f{:}\mathcal{B}_n.\downarrow\Sigma\alpha{:}\texttt{Free}.f = \alpha \in \mathcal{B}_n}$$

LS2 (discreteness) is valid

$$\overline{\Gamma \vdash \lambda\alpha, \beta.\_ : \Pi\alpha, \beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}) + (\neg\alpha = \beta \in \mathcal{B})}$$

Choice sequences in the Open Bar model

LS1 (density) is valid

$$\overline{\Gamma \vdash \lambda n, f.\star : \Pi n{:}\mathbb{N}.\Pi f{:}\mathcal{B}_n.\downarrow\Sigma\alpha{:}\texttt{Free}.f = \alpha \in \mathcal{B}_n}$$

LS2 (discreteness) is valid

$$\overline{\Gamma \vdash \lambda\alpha, \beta.\_ : \Pi\alpha, \beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}) + (\neg\alpha = \beta \in \mathcal{B})}$$

LS3 (Open Data) is valid

$$\begin{array}{l} \hline \Gamma \vdash \lambda\alpha, p.\star : \Pi\alpha{:}\texttt{Free}. \\ \qquad\qquad P(\alpha) \\ \qquad\qquad \rightarrow \downarrow\Sigma n{:}\mathbb{N}.\Pi\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow \downarrow P(\beta)) \end{array}$$

Open Data in the Open Bar model

$$\boldsymbol{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow {\downarrow}\boldsymbol{\Sigma}n{:}\mathbb{N}.\boldsymbol{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow {\downarrow}P(\beta))$$

Why is LS3 (Open Data) valid?

Open Data in the Open Bar model

$$\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow {\downarrow}\mathbf{\Sigma} n{:}\mathbb{N}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow {\downarrow}P(\beta))$$

Why is LS3 (Open Data) valid?

- no computational content: $\lambda\alpha, p.\star$
- we assume $P(\alpha)\texttt{@}w$, where w is the current world
- **within the metatheory we realize the modulus of continuity n with $|w|$ (w's depth)**

Open Data in the Open Bar model

$$\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \downarrow\mathbf{\Sigma}n{:}\mathbb{N}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow \downarrow P(\beta))$$

Why is LS3 (Open Data) valid?

- no computational content: $\lambda\alpha, p.\star$
- we assume $P(\alpha)@w$, where w is the current world
- **within the metatheory we realize the modulus of continuity n with $|w|$ (w's depth)**
- we get to assume that α and β have the same choices up to $|w|$ in some $w_1 \succeq w$, and we have to show $P(\beta)@w_1$
- there must be a world $w_1 \succeq w_0 \succeq w$ such that α and β have exactly the same choices in w_0
- by monotonicity: $P(\alpha)@w_0$
- we swap α and β: $P(\beta)@w_0$
- by monotonicity: $P(\beta)@w_1$

Open Data in the Open Bar model

$$\mathbf{\Pi}\alpha{:}\mathtt{Free}.P(\alpha) \rightarrow {\downarrow}\mathbf{\Sigma} n{:}\mathbb{N}.\mathbf{\Pi}\beta{:}\mathtt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow {\downarrow}P(\beta))$$

Can we validate a version with computational content?

Open Data in the Open Bar model

$$\mathbf{\Pi}\alpha{:}\mathtt{Free}.P(\alpha) \rightarrow \downarrow\mathbf{\Sigma} n{:}\mathbb{N}.\mathbf{\Pi}\beta{:}\mathtt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow \downarrow P(\beta))$$

Can we validate a version with computational content?

Can we compute n solely based on α and $P(\alpha)$?

Open Data in the Open Bar model

$$\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \downarrow\mathbf{\Sigma} n{:}\mathbb{N}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow \downarrow P(\beta))$$

Can we validate a version with computational content?
Can we compute n solely based on α and $P(\alpha)$?

At least:

$$\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \mathbf{\Sigma} n{:}\mathbb{N}_{\wr\downarrow}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_{\wr\downarrow n} \rightarrow \downarrow P(\beta))$$

Open Data in the Open Bar model

$$\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow {\downarrow}\mathbf{\Sigma}n{:}\mathbb{N}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow {\downarrow}P(\beta))$$

Can we validate a version with computational content?
Can we compute n solely based on α and $P(\alpha)$?

At least:

$$\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \mathbf{\Sigma}n{:}\mathbb{N}_{\rightsquigarrow}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_{\rightsquigarrow n} \rightarrow {\downarrow}P(\beta))$$

- We add an operator to the language to compute the depth of the current world:

$$t \in \texttt{Term} ::= \cdots \mid \texttt{wDepth}$$

Open Data in the Open Bar model

$$\Pi\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \downarrow\Sigma n{:}\mathbb{N}.\Pi\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow \downarrow P(\beta))$$

Can we validate a version with computational content?
Can we compute n solely based on α and $P(\alpha)$?

At least:

$$\Pi\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \Sigma n{:}\mathbb{N}_{\rightsquigarrow}.\Pi\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_{\rightsquigarrow n} \rightarrow \downarrow P(\beta))$$

- We add an operator to the language to compute the depth of the current world:

$$t \in \texttt{Term} ::= \cdots \mid \texttt{wDepth}$$

- We realize this formula using $\lambda\alpha, p.\langle\texttt{wDepth}, \ldots\rangle$

Open Data in the Open Bar model

2 variants of Open Data:

- $\mathbf{\Pi}\alpha{:}\mathtt{Free}.P(\alpha) \rightarrow \downarrow\mathbf{\Sigma}n{:}\mathbb{N}.\mathbf{\Pi}\beta{:}\mathtt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow \downarrow P(\beta))$
- $\mathbf{\Pi}\alpha{:}\mathtt{Free}.P(\alpha) \rightarrow \mathbf{\Sigma}n{:}\mathbb{N}_{\wr\downarrow}.\mathbf{\Pi}\beta{:}\mathtt{Free}.(\alpha = \beta \in \mathcal{B}_{\wr\downarrow n} \rightarrow \downarrow P(\beta))$

Open Data in the Open Bar model

2 variants of Open Data:

- $\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \downarrow\mathbf{\Sigma}n{:}\mathbb{N}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow \downarrow P(\beta))$
- $\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \mathbf{\Sigma}n{:}\mathbb{N}_{\rightsquigarrow}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_{\rightsquigarrow n} \rightarrow \downarrow P(\beta))$

Time squashing vs. space squashing:

- a member of $\mathbb{N}$ computes to the same value in all worlds
- a member of $\mathbb{N}_{\rightsquigarrow}$ can compute to $\neq$ values in $\neq$ worlds

Open Data in the Open Bar model

2 variants of Open Data:

- $\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \downarrow\mathbf{\Sigma} n{:}\mathbb{N}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_n \rightarrow \downarrow P(\beta))$
- $\mathbf{\Pi}\alpha{:}\texttt{Free}.P(\alpha) \rightarrow \mathbf{\Sigma} n{:}\mathbb{N}_{\wr}.\mathbf{\Pi}\beta{:}\texttt{Free}.(\alpha = \beta \in \mathcal{B}_{\wr n} \rightarrow \downarrow P(\beta))$

Time squashing vs. space squashing:

- a member of $\mathbb{N}$ computes to the same value in all worlds
- a member of $\mathbb{N}_{\wr}$ can compute to $\neq$ values in $\neq$ worlds

Also useful to assign types to references

Next. . .

- Can we still validate Bar Induction using such choice sequences?

Next...

- Can we still validate Bar Induction using such choice sequences?
- Can we still validate the Axiom of Choice using such choice sequences?

Next...

- Can we still validate Bar Induction using such choice sequences?
- Can we still validate the Axiom of Choice using such choice sequences?
- Can we validate continuity using references (similar to such choice sequences)?

Next. . .

- Can we still validate Bar Induction using such choice sequences?
- Can we still validate the Axiom of Choice using such choice sequences?
- Can we validate continuity using references (similar to such choice sequences)?
- . . .