Semantics of Higher-Order Logics

Christoph E. Benzmüller and Chad E. Brown

http://www.ags.uni-sb.de/~chris/

SEMHOL'06

Lecture Course at ESSLLI 2006, Malaga, Spain

Introduction

History

- Cantor's Set Theory – late 1800's
- Frege's Logic – late 1800's
- Russell's Paradox – 1902
- Zermelo's Axiomatic Set Theory – 1908
- Russell's Type Theory – 1908
- Church's Untyped λ-Calculus (Computation) – 1930's
- Church's Type Theory – HOL (Mathematics) – 1940
- Henkin Models and Completeness – 1950
- Cut-Elimination (Takahashi, Prawitz, Andrews) – 1967-1972
- Theorem Proving: Isabelle-HOL, HOL4, HOL-light, TPS, LEO – 1980's - today

Cantor's Set Theory

Sets:

$$\{x|\varphi(x)\}$$

$$a \in \{x|\varphi(x)\} \Leftrightarrow \varphi(a)$$

Power Set $\mathcal{P}(A)$ of A:

$$\{X|X \subseteq A\}$$

Cantor's Theorem

- **Cantor's Theorem** states that the powerset of A is always bigger than A.

Cantor's Theorem

- **Cantor's Theorem** states that the powerset of A is always bigger than A.
- Two versions of Cantor's Theorem are the surjective version and the injective version.

Cantor's Theorem

- **Cantor's Theorem** states that the powerset of A is always bigger than A.
- Two versions of Cantor's Theorem are the surjective version and the injective version.
- The **surjective Cantor Theorem** states that there does not exist a surjection from A onto $\mathcal{P}(\mathrm{A})$.

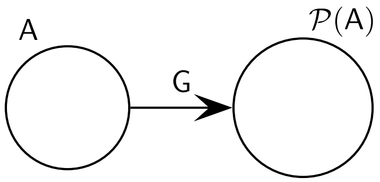
Cantor's Theorem

- **Cantor's Theorem** states that the powerset of A is always bigger than A.
- Two versions of Cantor's Theorem are the surjective version and the injective version.
- The **surjective Cantor Theorem** states that there does not exist a surjection from A onto $\mathcal{P}(\mathrm{A})$.
- The **injective Cantor Theorem** states that there does not exist an injection from $\mathcal{P}(\mathrm{A})$ into A.

Surjective Cantor Theorem

Cantor's Thm 1: There is no surjection from A onto $\mathcal{P}(A)$.

Proof:

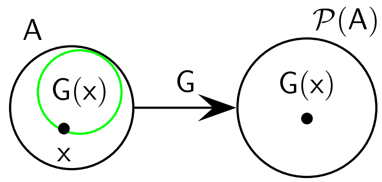


- Suppose $G : A \longrightarrow \mathcal{P}(A)$ is a surjection.

Surjective Cantor Theorem

Cantor's Thm 1: There is no surjection from A onto $\mathcal{P}(A)$.

Proof:

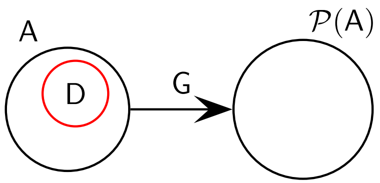


- Note for $x \in A$, $G(x) \subseteq A$.

Surjective Cantor Theorem

Cantor's Thm 1: There is no surjection from A onto $\mathcal{P}(A)$.

Proof:

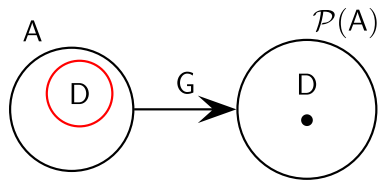


- Let D be the diagonal set $\{x \in A | x \notin G(x)\}$.

Surjective Cantor Theorem

Cantor's Thm 1: There is no surjection from A onto $\mathcal{P}(A)$.

Proof:

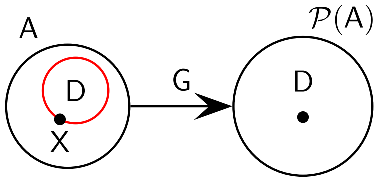


- Let D be the diagonal set $\{x \in A | x \notin G(x)\}$.
- $D \in \mathcal{P}(A)$.

Surjective Cantor Theorem

Cantor's Thm 1: There is no surjection from A onto $\mathcal{P}(A)$.

Proof:

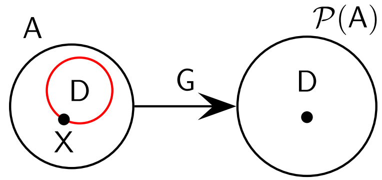


- Let D be the diagonal set $\{x \in A | x \notin G(x)\}$.
- Since G is surjective, $G(X) = D$ for some $X \in A$.

Surjective Cantor Theorem

Cantor's Thm 1: There is no surjection from A onto $\mathcal{P}(A)$.

Proof:



- Let D be the diagonal set $\{x \in A | x \notin G(x)\}$.
- $X \in G(X)$ iff $X \in D$ iff $X \notin G(X)$. *Contradiction!*

Russell's Paradox

Reconsider Diagonal Set:

$$D := \{x \in A | x \notin G(x)\}$$

Easy Modification Gives Russell Set:

$$R := \{x | x \notin x\}$$

Russell's Paradox: $R \in R \Leftrightarrow R \notin R$

This shows Cantor's Naive Set Theory is Inconsistent

And Frege's Logic is Inconsistent

Resolving Russell's Paradox

Zermelo (later Zermelo-Fraenkel) Axiomatic Set Theory
(Sets are not "too big")

Russell's Type Theory (Syntactic Hierarchy of Objects)

Using Simply Typed λ-Calculus,

Can Formulate Church's Type Theory

Church's Untyped λ-Calculus

Pure Calculus of Functions

$$\mathsf{x} \quad | \quad [\lambda \mathsf{x} \mathbf{M}] \quad | \quad [\mathbf{F}\ \mathbf{A}]$$

β-reduction: $[[\lambda \mathsf{x} \mathbf{M}]\ \mathbf{A}]$ reduces to $[\mathbf{A}/\mathsf{x}]\mathbf{M}$

Foundation for Computation:
Recursive Functions $\cong$ λ-Definable Functions

Problematic for Mathematics (but still hope, see Map Calculus)

Church's Type Theory (HOL)

1940:

- Simple Types (including type o of "truth values")
- Simply Typed λ-Terms (including logical constants)
- Deductive System
- Sufficient for Much of Mathematics

Simple Types

	o	(truth values)
Simple Types $\mathcal{T}$:	ι	(individuals)
	$(\alpha \to \beta)$	(functions from α to β)

$(\alpha \longrightarrow \beta)$ is sometimes written $(\beta\alpha)$

$(\alpha \to \beta \to \gamma)$ abbreviates $(\alpha \to (\beta \to \gamma))$

Simple Types

	o	(truth values)
Simple Types $\mathcal{T}$:	ι	(individuals)
	$(\alpha \to \beta)$	(functions from α to β)

$\mathcal{T}$ is a freely generated, inductive set.

Induction on Types: We can prove a property $\varphi(\alpha)$ holds for all types α by proving

- $\varphi(\mathsf{o})$
- $\varphi(\iota)$
- If $\varphi(\alpha)$ and $\varphi(\beta)$, then $\varphi(\alpha \to \beta)$.

Simple Types

	o	(truth values)
Simple Types $\mathcal{T}$:	ι	(individuals)
	$(\alpha \to \beta)$	(functions from α to β)

Recursion on Types: We can uniquely define a family $\mathcal{D}_\alpha$ for $\alpha \in \mathcal{T}$ by specifying:

- $\mathcal{D}_o$
- $\mathcal{D}_\iota$
- A rule for forming $\mathcal{D}_{\alpha\to\beta}$ given $\mathcal{D}_\alpha$ and $\mathcal{D}_\beta$.

A Standard Frame

$$\begin{aligned}\mathcal{D}_o &= \{\mathtt{T},\mathtt{F}\}.\\ \mathcal{D}_\iota &= \mathbf{N} \text{ (natural numbers).}\\ \mathcal{D}_{\alpha\to\beta} &= \mathcal{D}_\beta^{\mathcal{D}_\alpha}\text{, all functions from } \mathcal{D}_\alpha \text{ to } \mathcal{D}_\beta.\end{aligned}$$

A Standard Frame

$$\begin{aligned}\mathcal{D}_o &= \{\mathtt{T},\mathtt{F}\}.\\ \mathcal{D}_\iota &= \mathbf{N} \text{ (natural numbers).}\\ \mathcal{D}_{\alpha\to\beta} &= \mathcal{D}_\beta^{\mathcal{D}_\alpha}\text{, all functions from } \mathcal{D}_\alpha \text{ to } \mathcal{D}_\beta.\end{aligned}$$

$\mathcal{D}_{\iota\to o} \cong \mathcal{P}(\mathbf{N})$:

$X \subseteq \mathbf{N}$ induces $\chi_X \in \mathcal{D}_{\iota\to o}$ (characteristic function)

$$\chi_X(x) := \begin{cases} \mathtt{T} & \text{if } x \in X \\ \mathtt{F} & \text{if } x \notin X \end{cases}$$

Every $f \in \mathcal{D}_{\iota\to o}$ is χ_X where

$$X := \{x \in \mathcal{D}_\iota | f(x) = \mathtt{T}\}$$

A Standard Frame

$$\begin{aligned}\mathcal{D}_o &= \{\mathtt{T},\mathtt{F}\}.\\ \mathcal{D}_\iota &= \mathbf{N} \text{ (natural numbers).}\\ \mathcal{D}_{\alpha\to\beta} &= \mathcal{D}_\beta^{\mathcal{D}_\alpha}\text{, all functions from } \mathcal{D}_\alpha \text{ to } \mathcal{D}_\beta.\end{aligned}$$

$\mathcal{D}_{\iota\to\iota\to o} \cong \mathcal{P}(\mathbf{N}\times\mathbf{N})$: Binary relations on $\mathbf{N}$

$\mathcal{D}_{(\iota\to o)\to o} \cong \mathcal{P}(\mathcal{P}(\mathbf{N}))$

Standard Frames

$\mathcal{D}_o$ = any nonempty set

$\mathcal{D}_\iota$ = any nonempty set

$\mathcal{D}_{\alpha\to\beta}$ = $(\mathcal{D}_\beta)^{\mathcal{D}_\alpha}$, all functions from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$.

Standard Frames are Determined by Domains of Base Type: If $\mathcal{D}$ and $\mathcal{E}$ are standard frames, $\mathcal{D}_o = \mathcal{E}_o$, and $\mathcal{D}_\iota = \mathcal{E}_\iota$, then $\mathcal{D} = \mathcal{E}$.

Proof: Induction on types.

Simply Typed Lambda Terms

- Σ - typed logical constants
- $\mathcal{V}$ - typed variables (each $\mathcal{V}_\alpha$ infinite)
- $\mathcal{P}$ - typed parameters (each $\mathcal{P}_\alpha$ of same infinite cardinality)

Simply Typed Lambda Terms

- $\mathsf{x}_\alpha \in \mathit{wff}_\alpha(\Sigma)$ for each variable $\mathsf{x}_\alpha \in \mathcal{V}_\alpha$.
- $\mathsf{W}_\alpha \in \mathit{wff}_\alpha(\Sigma)$ for each parameter $\mathsf{W}_\alpha \in \mathcal{P}_\alpha$.
- $\mathsf{c}_\alpha \in \mathit{wff}_\alpha(\Sigma)$ for each constant $\mathsf{c}_\alpha \in \Sigma_\alpha$.
- $[\mathbf{F}_{\alpha\to\beta}\,\mathbf{A}_\alpha] \in \mathit{wff}_\beta(\Sigma)$ for each $\mathbf{F} \in \mathit{wff}_{\alpha\to\beta}(\Sigma)$ and $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$.
- $[\lambda \mathsf{x}_\alpha \mathbf{B}_\beta] \in \mathit{wff}_{\alpha\to\beta}(\Sigma)$ for each variable $\mathsf{x}_\alpha \in \mathcal{V}_\alpha$ and $\mathbf{B} \in \mathit{wff}_\beta(\Sigma)$.

Simply Typed Lambda Terms

- $\mathbf{Free}(\mathbf{A}_\alpha) \subset \mathcal{V}$ - set of free variables in $\mathbf{A}$
- A term is $\mathbf{A}_\alpha$ *closed* if there are no free variables in $\mathbf{A}$.
- $\mathit{cwff}_\alpha(\Sigma)$ (or cwff_α) := set of all closed terms of type α.

Logical Constants

Some logical constants we will consider:

$\neg_{o\to o}$	negation
$\vee_{o\to o\to o}$	disjunction
$\wedge_{o\to o\to o}$	conjunction
$=^{\alpha}_{\alpha\to\alpha\to o}$	equality at type α
$\Pi^{\alpha}_{(\alpha\to o)\to o}$	universal quantification over type α
$\Sigma^{\alpha}_{(\alpha\to o)\to o}$	existential quantification over type α

Intuition: $[\Sigma^{\alpha} \blacksquare \lambda x_{\alpha}\, \mathbf{C}_o]$ maps to true iff $\{x_{\alpha} \mid \mathbf{C}\}$ is nonempty.

Peano Arithmetic

Easy to Encode Peano's Axioms with ι as $\mathbf{N}$,
0_{ι} a parameter and $S_{\iota\to\iota}$ a parameter

1. Zero is a natural number.
   0 has type ι
2. n natural number $\Rightarrow$ successor of n is a natural number
   $[S\, \mathbf{N}]$ has type ι for any term $\mathbf{N}_{\iota}$
3. No successor is zero.
   $\forall n_{\iota} \neg[[S\, n] = 0]$
4. The successor function is injective.

5. Induction:

Peano Arithmetic

Easy to Encode Peano's Axioms with ι as $\mathbf{N}$,
0_{ι} a parameter and $S_{\iota\to\iota}$ a parameter

1. Zero is a natural number.
   0 has type ι
2. n natural number $\Rightarrow$ successor of n is a natural number
   $[S\, \mathbf{N}]$ has type ι for any term $\mathbf{N}_{\iota}$
3. No successor is zero.
   $[\Pi^{\iota}_{(\iota\to o)\to o}[\lambda n_{\iota}[\neg_{o\to o}[=^{\iota}_{\iota\to\iota\to o}[S_{\iota\to\iota}\, n]\, 0_{\iota}]]]]_o$
4. The successor function is injective.

5. Induction:

Peano Arithmetic

Easy to Encode Peano's Axioms with ι as $\mathbf{N}$,
0_{ι} a parameter and $S_{\iota\to\iota}$ a parameter

1. Zero is a natural number.
   0 has type ι
2. n natural number $\Rightarrow$ successor of n is a natural number
   $[S\, \mathbf{N}]$ has type ι for any term $\mathbf{N}_{\iota}$
3. No successor is zero.
   $\forall n_{\iota} \neg[[S\, n] = 0]$
4. The successor function is injective.
   $\forall n_{\iota} \forall m_{\iota}[[[S\, n] = [S\, m]] \supset n = m]$
5. Induction:

Peano Arithmetic

Easy to Encode Peano's Axioms with ι as $\mathbf{N}$,
0_ι a parameter and $S_{\iota\to\iota}$ a parameter

1. Zero is a natural number.
   0 has type ι
2. n natural number $\Rightarrow$ successor of n is a natural number
   $[S\ \mathbf{N}]$ has type ι for any term $\mathbf{N}_\iota$
3. No successor is zero.
   $\forall n_\iota \neg[[S\ n] = 0]$
4. The successor function is injective.
   $\forall n_\iota \forall m_\iota [[[S\ n] = [S\ m]] \supset n = m]$
5. Induction: $\forall p_{\iota\to o}[[p\ 0] \wedge [\forall n_\iota[[p\ n] \supset [p\ [S\ n]]]] \supset [\forall n_\iota\ [p\ n]]]$

Incompleteness wrt Standard Frames

Only ONE standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ satisfies Peano: $\mathcal{D}_\iota = \mathbf{N}$

Suppose we have a recursively axiomatizable deduction system $\vdash$ for HOL sound and complete for standard models with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$.

Gödel construction gives: $\mathbf{G}_o$

$\mathbf{G}$ evaluates to $\mathtt{T}$ in standard frame $\mathcal{D}$ above $\Leftrightarrow \not\vdash [PA \supset \mathbf{G}]$

$\vdash [PA \supset \mathbf{G}] \Rightarrow_{\text{Soundness}} \mathbf{G}$ evaluates to $\mathtt{T}$ in $\mathcal{D} \Rightarrow \not\vdash [PA \supset \mathbf{G}]$

$\not\vdash [PA \supset \mathbf{G}] \Rightarrow \mathbf{G}$ evaluates to $\mathtt{T}$ in $\mathcal{D} \Rightarrow_{\text{Completeness}} \vdash [PA \supset \mathbf{G}]$

Incompleteness wrt Standard Frames

Only ONE standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ satisfies Peano: $\mathcal{D}_\iota = \mathbf{N}$

Suppose we have a recursively axiomatizable deduction system $\vdash$ for HOL sound and complete for standard models with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$.

Gödel construction gives: $\mathbf{G}_o$

$\mathbf{G}$ evaluates to $\mathtt{T}$ in standard frame $\mathcal{D}$ above $\Leftrightarrow \not\vdash [PA \supset \mathbf{G}]$

There is no recursively axiomatizable deduction system for HOL sound and complete wrt standard models.

Frames in General

$\mathcal{D}_o$ = any nonempty set

$\mathcal{D}_\iota$ = any nonempty set

$\mathcal{D}_{\alpha\to\beta} \subseteq (\mathcal{D}_\beta)^{\mathcal{D}_\alpha}$ (maybe not all functions)

Frames are NOT Determined by Domains of Base Type.

Henkin Completeness (1950): Church's Deductive System is Complete wrt a Class of General Frames ("Henkin Models")

Theorem Proving in HOL

Interactive Systems for Constructing Formal Theories (these use extensions of Church's Type Theory):

- Isabelle-HOL
- HOL-light
- HOL4

Systems Performing Automated Search for Proofs in (Fragments of) Church's Type Theory:

- TPS
- LEO

Theorem Proving: Extensionality

Consider $[A_o \wedge B_o \wedge [Q_{o\to o} A]] \supset [Q B]$.

Theorem? Yes, assuming Boolean Extensionality.
Idea: A and B true implies A and B are equal.

Theorem Proving: Extensionality

Consider $[A_o \wedge B_o \wedge [Q_{o\to o} A]] \supset [Q B]$.

Theorem? Yes, assuming Boolean Extensionality.
Idea: A and B true implies A and B are equal.

Automatic Search? Clauses to Refute:
A
B
$[Q A]$
$\neg[Q B]$
What to resolve?

Theorem Proving: Extensionality

Consider $[A_o \wedge B_o \wedge [Q_{o\to o} A]] \supset [Q B]$.

Theorem? Yes, assuming Boolean Extensionality.
Idea: A and B true implies A and B are equal.

Automatic Search? Clauses to Refute:
A
B
$[Q A]$
$\neg[Q B]$
What to resolve?
None Unify Syntactically.

Theorem Proving: Extensionality

Consider $[A_o \wedge B_o \wedge [Q_{o\to o} A]] \supset [Q\,B]$.

Theorem? Yes, assuming Boolean Extensionality.
Idea: A and B true implies A and B are equal.

Automatic Search? Clauses to Refute:
A
B
$[Q\,A]$
$\neg[Q\,B]$
What to resolve?
None Unify Syntactically.
Idea: Resolve $[Q\,A]$ and $\neg[Q\,B]$, then prove $A = B$

Theorem Proving: Extensionality

There are Similar Examples for Functional Extensionality

TPS traditionally searches without extensionality.

TPS Could Not Prove Such Examples

TPS Was Not "Henkin Complete" (but maybe wrt other model classes)?

LEO (1999) introduced Search with Extensionality

Theorem Proving: Logical Constants

During Automated Search, must consider "Set Instantiations" involving logical constants.

Example, Diagonal Set in Injective Cantor Theorem is of the form:

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y =^{\iota} H(X) \wedge \neg[X[HX]]]]$$

How can an Automated Prover discover this?

Primitive Substitutions. (Guessing)

Instantiation during Search

$$w_{(\iota\to o)\to o}$$

Initially a Variable

Theorem Proving: Logical Constants

During Automated Search, must consider "Set Instantiations" involving logical constants.

Example, Diagonal Set in Injective Cantor Theorem is of the form:

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y =^{\iota} H(X) \wedge \neg[X[HX]]]]$$

How can an Automated Prover discover this?

Primitive Substitutions. (Guessing)

Instantiation during Search

$$[\lambda Y_{\iota\to o}[w\,Y]]$$

Initially a Variable (η-long form)

Theorem Proving: Logical Constants

During Automated Search, must consider “Set Instantiations” involving logical constants.

Example, Diagonal Set in Injective Cantor Theorem is of the form:

$$[\lambda Y_{\iota\rightarrow o}.\exists X_{\iota\rightarrow o}[Y =^{\iota} H(X) \wedge \neg[X[HX]]]]$$

How can an Automated Prover discover this?

Primitive Substitutions. (Guessing)

Instantiation during Search

$$[\lambda Y_{\iota\rightarrow o}.\exists X_{\iota\rightarrow o}[w^1\, X\, Y]]$$

Guess $\Sigma^{\iota\rightarrow o}$ at the head (guessing Σ and type!)

Theorem Proving: Logical Constants

During Automated Search, must consider “Set Instantiations” involving logical constants.

Example, Diagonal Set in Injective Cantor Theorem is of the form:

$$[\lambda Y_{\iota\rightarrow o}.\exists X_{\iota\rightarrow o}[Y =^{\iota} H(X) \wedge \neg[X[HX]]]]$$

How can an Automated Prover discover this?

Primitive Substitutions. (Guessing)

Instantiation during Search

$$[\lambda Y_{\iota\rightarrow o}.\exists X_{\iota\rightarrow o}[[w^2\, X\, Y] \wedge [w^3\, X\, Y]]]$$

Guess $\wedge$

Theorem Proving: Logical Constants

During Automated Search, must consider “Set Instantiations” involving logical constants.

Example, Diagonal Set in Injective Cantor Theorem is of the form:

$$[\lambda Y_{\iota\rightarrow o}.\exists X_{\iota\rightarrow o}[Y =^{\iota} H(X) \wedge \neg[X[HX]]]]$$

How can an Automated Prover discover this?

Primitive Substitutions. (Guessing)

Instantiation during Search

$$[\lambda Y_{\iota\rightarrow o}.\exists X_{\iota\rightarrow o}[[w^2\, X\, Y] \wedge \neg[w^4\, X\, Y]]]$$

Guess $\neg$ for w^3

Theorem Proving: Logical Constants

During Automated Search, must consider “Set Instantiations” involving logical constants.

Example, Diagonal Set in Injective Cantor Theorem is of the form:

$$[\lambda Y_{\iota\rightarrow o}.\exists X_{\iota\rightarrow o}[Y =^{\iota} H(X) \wedge \neg[X[HX]]]]$$

How can an Automated Prover discover this?

Primitive Substitutions. (Guessing)

Instantiation during Search

$$[\lambda Y_{\iota\rightarrow o}.\exists X_{\iota\rightarrow o}[[[w^5\, X\, Y] =^{\iota} [w^6\, X\, Y]] \wedge \neg[w^4\, X\, Y]]]$$

Guess $=^{\iota}$ for w^2 (guessing type!)

Theorem Proving: Logical Constants

During Automated Search, must consider "Set Instantiations" involving logical constants.

Example, Diagonal Set in Injective Cantor Theorem is of the form:

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y =^{\iota} H(X) \wedge \neg[X[HX]]]]$$

How can an Automated Prover discover this?

Primitive Substitutions. (Guessing)

Instantiation during Search

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[[[w^5\, X\, Y] =^{\iota} [w^6\, X\, Y]] \wedge \neg[w^4\, X\, Y]]]$$

Now we have the right logical form

Theorem Proving: Logical Constants

During Automated Search, must consider "Set Instantiations" involving logical constants.

Example, Diagonal Set in Injective Cantor Theorem is of the form:

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y =^{\iota} H(X) \wedge \neg[X[HX]]]]$$

How can an Automated Prover discover this?

Primitive Substitutions. (Guessing)

Instantiation during Search

Not Realistic During Search. What if we don't bother?

Is there an appropriate semantics without all logical constants?

Coming Attractions

- Semantics without all Logical Constants
- Semantics without full Extensionality

Cantor's Theorem Revisited

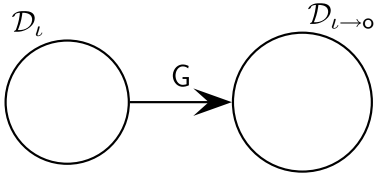
Cantor's Theorem

- **Cantor's Theorem** states that the powerset of A is always bigger than A.
- In the simply typed setting, the type ι will correspond to the set A and the type $\iota \to o$ will correspond to the set $\mathcal{P}(A)$.

Surjective Cantor in Standard Frames

Proposition: Suppose $\mathcal{D}$ is a standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$. There is no surjection in $\mathcal{D}_{\iota \to (\iota \to o)}$.

Proof:

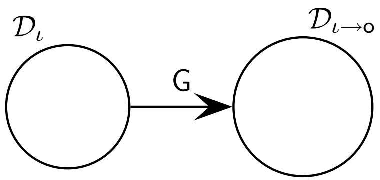


- Suppose $G \in \mathcal{D}_{\iota \to \iota \to o}$ is a surjection.

Surjective Cantor in Standard Frames

Proposition: Suppose $\mathcal{D}$ is a standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$. There is no surjection in $\mathcal{D}_{\iota \to (\iota \to o)}$.

Proof:

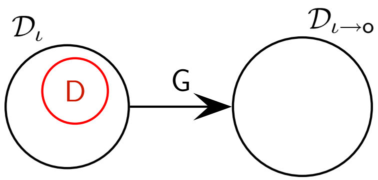


- That is, $G : \mathcal{D}_\iota \to \mathcal{D}_{\iota \to o}$ is a surjection.

Surjective Cantor in Standard Frames

Proposition: Suppose $\mathcal{D}$ is a standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$. There is no surjection in $\mathcal{D}_{\iota \to (\iota \to o)}$.

Proof:

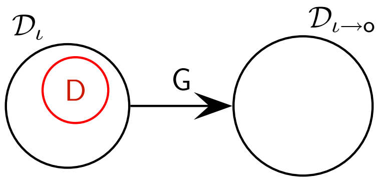


- Let D be the diagonal set $\{x \in \mathcal{D}_\iota | G(x)(x) = \mathtt{F}\}$.

Surjective Cantor in Standard Frames

Proposition: Suppose $\mathcal{D}$ is a standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$. There is no surjection in $\mathcal{D}_{\iota\to(\iota\to o)}$.

Proof:

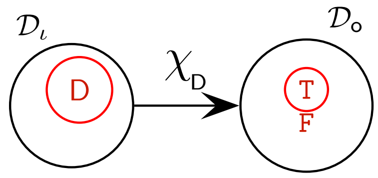


- Let D be the diagonal set $\{x \in \mathcal{D}_\iota | G(x)(x) = \mathtt{F}\}$.
- Intuitively, $G(x)(x) = \mathtt{F}$ means "$x \notin G(x)$."

Surjective Cantor in Standard Frames

Proposition: Suppose $\mathcal{D}$ is a standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$. There is no surjection in $\mathcal{D}_{\iota\to(\iota\to o)}$.

Proof:

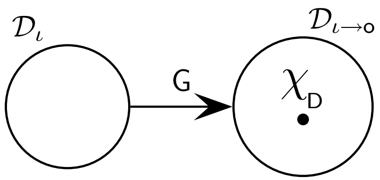


- Let D be the diagonal set $\{x \in \mathcal{D}_\iota | G(x)(x) = \mathtt{F}\}$.
- Consider $\chi_D : \mathcal{D}_\iota \to \mathcal{D}_o$ defined by $\chi_D(x) := \mathtt{T}$ iff $x \in D$.

Surjective Cantor in Standard Frames

Proposition: Suppose $\mathcal{D}$ is a standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$. There is no surjection in $\mathcal{D}_{\iota\to(\iota\to o)}$.

Proof:

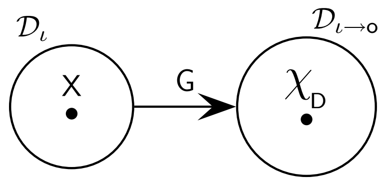


- Let D be the diagonal set $\{x \in \mathcal{D}_\iota | G(x)(x) = \mathtt{F}\}$.
- Since $\mathcal{D}$ is standard, $\chi_D \in \mathcal{D}_{\iota\to o}$.

Surjective Cantor in Standard Frames

Proposition: Suppose $\mathcal{D}$ is a standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$. There is no surjection in $\mathcal{D}_{\iota\to(\iota\to o)}$.

Proof:

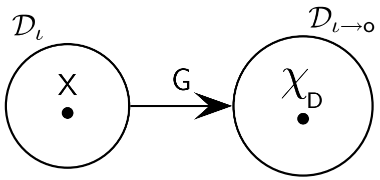


- Let D be the diagonal set $\{x \in \mathcal{D}_\iota | G(x)(x) = \mathtt{F}\}$.
- Since G is surjective, $G(X) = \chi_D$ for some $X \in \mathcal{D}_\iota$.

Surjective Cantor in Standard Frames

Proposition: Suppose $\mathcal{D}$ is a standard frame with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$. There is no surjection in $\mathcal{D}_{\iota\to(\iota\to o)}$.

Proof:



Contradiction!

- Let D be the diagonal set $\{x \in \mathcal{D}_\iota | G(x)(x) = \mathtt{F}\}$.
- $G(X)(X) = \mathtt{T}$ iff $\chi_D(X) = \mathtt{T}$ iff $X \in D$ iff $G(X)(X) = \mathtt{F}$

Surjective Cantor in Frames

Does the surjective Cantor Theorem also hold in more general frames?

Surjective Cantor in Frames

Does the surjective Cantor Theorem also hold in more general frames?

Not Always

Surjective Cantor in Frames

Does the surjective Cantor Theorem also hold in more general frames?

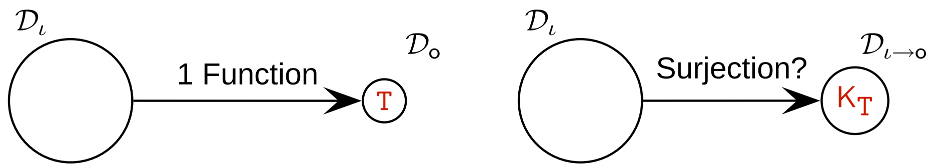
Not Always

Let $\mathcal{D}$ be a frame.

Defn: We say Surjective Cantor Holds in $\mathcal{D}$ if there is no surjection in $\mathcal{D}_{\iota\to\iota\to o}$. Otherwise, we say Surjective Cantor Fails.

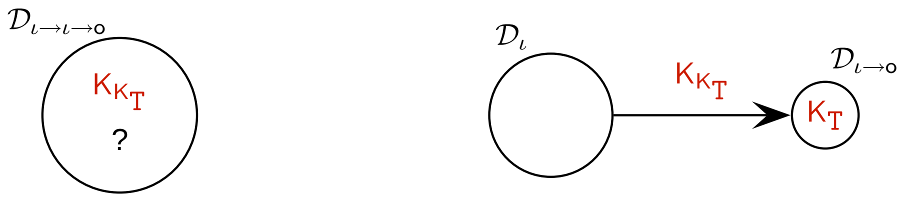
Example Frame 1

- Let $\mathcal{D}$ be a standard frame with $\mathcal{D}_o = \{\mathtt{T}\}$.
- $\mathcal{D}_{\iota\to o}$ has exactly one element



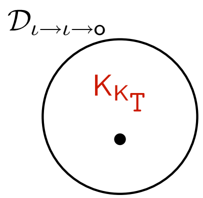
Example Frame 1

- Let $\mathcal{D}$ be a standard frame with $\mathcal{D}_o = \{\mathtt{T}\}$.
- $\mathcal{D}_{\iota\to o}$ has exactly one element
- There is a surjection from $\mathcal{D}_\iota$ onto $\mathcal{D}_{\iota\to o}$.



Example Frame 1

- Let $\mathcal{D}$ be a standard frame with $\mathcal{D}_o = \{\mathtt{T}\}$.
- $\mathcal{D}_{\iota\to o}$ has exactly one element
- There is a surjection from $\mathcal{D}_\iota$ onto $\mathcal{D}_{\iota\to o}$.
- Since $\mathcal{D}$ is standard, the surjection is in $\mathcal{D}_{\iota\to\iota\to o}$.

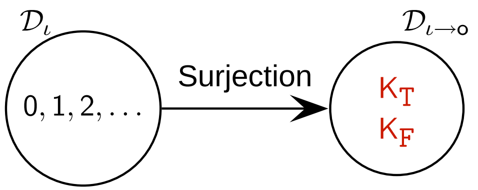


Example Frame 1

- Let $\mathcal{D}$ be a standard frame with $\mathcal{D}_o = \{\mathtt{T}\}$.
- $\mathcal{D}_{\iota\to o}$ has exactly one element
- There is a surjection from $\mathcal{D}_\iota$ onto $\mathcal{D}_{\iota\to o}$.
- Since $\mathcal{D}$ is standard, the surjection is in $\mathcal{D}_{\iota\to\iota\to o}$.
- **Surjective Cantor Fails in this frame. For Cantor, it is vital $\mathcal{D}_o$ has at least two elements.**

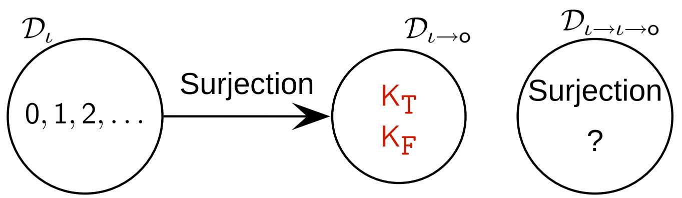
Example Frame 2

- Let $\mathcal{D}$ be defined by $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota := \mathbf{N}$,
- $\mathcal{D}_{\alpha\to\beta}$ to be the set of *constant* functions from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$.
- In this case $\mathcal{D}_{\iota\to o}$ contains two elements (the constant $\mathtt{T}$ function and the constant $\mathtt{F}$ function).
- Clearly, there is a surjection from $\mathbf{N}$ to the two element set $\mathcal{D}_{\iota\to o}$.



Example Frame 2

- Let $\mathcal{D}$ be defined by $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota := \mathbf{N}$,
- $\mathcal{D}_{\alpha\to\beta}$ to be the set of *constant* functions from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$.
- In this case $\mathcal{D}_{\iota\to o}$ contains two elements (the constant $\mathtt{T}$ function and the constant $\mathtt{F}$ function).
- Clearly, there is a surjection from $\mathbf{N}$ to the two element set $\mathcal{D}_{\iota\to o}$.



Example Frame 2

- Let $\mathcal{D}$ be defined by $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota := \mathbf{N}$,
- $\mathcal{D}_{\alpha\to\beta}$ to be the set of *constant* functions from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$.
- In this case $\mathcal{D}_{\iota\to o}$ contains two elements (the constant $\mathtt{T}$ function and the constant $\mathtt{F}$ function).
- Clearly, there is a surjection from $\mathbf{N}$ to the two element set $\mathcal{D}_{\iota\to o}$.
- *No such surjection is a constant function.*
- No such surjections are in $\mathcal{D}_{\iota\to\iota\to o}$.
- $\mathcal{D}_{\iota\to\iota\to o}$ is also a two element set, neither surjective
- **Surjective Cantor Holds in this frame.**

Example Frame 3

- Let $\mathcal{D}$ be defined by $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota := \mathbf{N}$,
- $\mathcal{D}_{\iota\to o}$ to be the set of *constant* functions from $\mathcal{D}_\iota$ to $\mathcal{D}_o$,
- $\mathcal{D}_{\alpha\to\beta}$ to be the set of *all* functions from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$ if $(\alpha \to \beta) \neq (\iota \to o)$.
- $\mathcal{D}_{\iota\to o}$ contains two elements,
- There is a surjection from $\mathbf{N}$ onto $\mathcal{D}_{\iota\to o}$.
- In this case, $\mathcal{D}_{\iota\to\iota\to o}$ is the set of all functions, so any such surjection is in $\mathcal{D}_{\iota\to\iota\to o}$.
- **Surjective Cantor Fails in this frame.**

Example Frame 3

- Let $\mathcal{D}$ be defined by $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota := \mathbf{N}$,
- $\mathcal{D}_{\iota\to o}$ to be the set of *constant* functions from $\mathcal{D}_\iota$ to $\mathcal{D}_o$,
- $\mathcal{D}_{\alpha\to\beta}$ to be the set of *all* functions from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$ if $(\alpha \to \beta) \neq (\iota \to o)$.
- $\mathcal{D}_{\iota\to o}$ contains two elements,
- There is a surjection from $\mathbf{N}$ onto $\mathcal{D}_{\iota\to o}$.
- In this case, $\mathcal{D}_{\iota\to\iota\to o}$ is the set of all functions, so any such surjection is in $\mathcal{D}_{\iota\to\iota\to o}$.
- **Surjective Cantor Fails in this frame.**
- HOWEVER, One cannot interpret typed λ-calculus in this frame. Consider $[\lambda g_{\iota\to\iota\to o} \lambda x_\iota\, [g\, x\, x]]$.

Interpreting Lambda Terms

Interpreting Terms in Standard Frames

- Let $\mathcal{D}$ be a standard frame.
- An *assignment* φ into $\mathcal{D}$ is $\varphi : \mathcal{V} \to \mathcal{D}$ typed function (so $\varphi(x_\alpha) \in \mathcal{D}_\alpha$).
- We use $\varphi, [a/x]$ to denote the assignment which agrees with φ except that $(\varphi, [a/x])(x_\alpha) := a$.
- An "interpretation" $\mathcal{I}$ of parameters and logical constants is a typed function $\mathcal{I} : (\mathcal{P} \cup \Sigma) \to \mathcal{D}$

Interpreting Terms in Standard Frames

Given $\mathcal{I} : (\mathcal{P} \cup \Sigma) \to \mathcal{D}$.
Define $\mathcal{E}_\varphi(\mathbf{M})$ by induction on $\mathbf{M}$:

- $\mathcal{E}_\varphi(x) := \varphi(x)$ for variables $x \in \mathcal{V}$.
- $\mathcal{E}_\varphi(c) := \mathcal{I}(c)$ for parameters or logical constants $c \in \mathcal{P} \cup \Sigma$.
- $\mathcal{E}_\varphi(\mathbf{F}_{\alpha\to\beta}\mathbf{A}_\alpha) := \mathcal{E}_\varphi(\mathbf{F})(\mathcal{E}_\varphi(\mathbf{A}))$ – using the fact that $\mathcal{E}_\varphi(\mathbf{F}_{\alpha\to\beta}) \in \mathcal{D}_{\alpha\to\beta}$ is a function from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$.
- Let $\mathcal{E}_\varphi([\lambda x_\alpha \mathbf{B}_\beta])$ be the function from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$ taking $a \in \mathcal{D}_\alpha$ to $\mathcal{E}_{\varphi,[a/x]}(\mathbf{B})$. This function is in $\mathcal{D}_{\alpha\to\beta}$ since $\mathcal{D}$ is standard.

$\mathcal{E}$ "evaluates" λ-terms.

Combinatory Frames

- To define $\mathcal{E}$ on λ-abstractions, we used $\mathcal{D}$ standard.
- Want weaker conditions on $\mathcal{D}$
- $\mathcal{D}$ *Combinatory*

Combinatory Formulas

- $\mathbf{K}_{\alpha\to\beta\to\alpha} := \lambda x_\alpha \lambda y_\beta\, x$
- $\mathbf{S}_{(\alpha\to\beta\to\gamma)\to(\alpha\to\beta)\to\alpha\to\gamma} := \lambda u_{\alpha\to\beta\to\gamma} \lambda v_{\alpha\to\beta} \lambda w_\alpha\, [u\, w\, [v\, w]]$

$\mathbf{SK}$-combinatory formulas contain every $\mathbf{K}$ and $\mathbf{S}$, every constant, every parameter, every variable, and are closed under application.

Combinatory Frames

Frame $\mathcal{D}$ is combinatory if:
for all types α, β and γ
there exist elements $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$ and
$\mathtt{s} \in \mathcal{D}_{(\alpha\to\beta\to\gamma)\to(\alpha\to\beta)\to\alpha\to\gamma}$ such that

- $\mathtt{k}(a)(b) = a$ for every $a \in \mathcal{D}_\alpha$ and $b \in \mathcal{D}_\beta$, and
- $\mathtt{s}(g)(f)(a) = g(a)(f(a))$ for every $a \in \mathcal{D}_\alpha$, $f \in \mathcal{D}_{\alpha\to\beta}$ and $g \in \mathcal{D}_{\alpha\to\beta\to\gamma}$.

Evaluations in Combinatory Frames

Proposition: Let $\mathcal{D}$ be a combinatory frame.
Let $\mathcal{I} : (\mathcal{P} \cup \Sigma) \to \mathcal{D}$ be an interpretation of constants and parameters.
There is a function $\mathcal{E}$ such that for every assignment φ and term $\mathbf{M}_\beta$, $\mathcal{E}_\varphi(\mathbf{M}_\beta) \in \mathcal{D}_\beta$ and the following hold:

- $\mathcal{E}_\varphi(x) = \varphi(x)$ for variables $x \in \mathcal{V}$.
- $\mathcal{E}_\varphi(c) = \mathcal{I}(c)$ for parameters or logical constants c.
- $\mathcal{E}_\varphi(\mathbf{F}_{\alpha\to\beta}\mathbf{A}_\alpha) = \mathcal{E}_\varphi(\mathbf{F})(\mathcal{E}_\varphi(\mathbf{A}))$
- $\mathcal{E}_\varphi([\lambda x_\alpha \mathbf{B}_\beta])$ is the function from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$ taking $a \in \mathcal{D}_\alpha$ to $\mathcal{E}_{\varphi,[a/x]}(\mathbf{B})$.

Frames Realizing Logical Constants

$$\mathcal{D}_o \subseteq \{\mathtt{T},\mathtt{F}\} \text{ (nonempty)}$$
$$\mathcal{D}_\iota = \text{any nonempty set}$$
$$\mathcal{D}_{\alpha\to\beta} \subseteq (\mathcal{D}_\beta)^{\mathcal{D}_\alpha} \text{ (maybe not all functions)}$$

We may want $\mathcal{D}$ to include interpretations for logical constants.

Defn: We say $\mathcal{D}$ realizes a logical constant c_α if there is an *appropriate* interpretation for c in $\mathcal{D}_\alpha$.

Examples:

- $\mathcal{D}$ realizes $\neg$ if the negation function $\neg : \mathcal{D}_o \longrightarrow \mathcal{D}_o$ is in $\mathcal{D}_{o\to o}$.
- $\mathcal{D}$ realizes Π^α if there is some function $\pi \in \mathcal{D}_{(\alpha\to o)\to o}$ such that $\pi(f) = \mathtt{T}$ iff $f(a) = \mathtt{T}$ for every $a \in \mathcal{D}_\alpha$.

Properties for Logical Constants

prop.	where	holds when			for all
$\mathfrak{L}_\neg(n)$	$n \in \mathcal{D}_{o\to o}$	$n(a) = \mathtt{T}$	iff	$a = \mathtt{F}$	$a \in \mathcal{D}_o$
$\mathfrak{L}_\vee(d)$	$d \in \mathcal{D}_{o\to o\to o}$	$d(a)(b) = \mathtt{T}$	iff	$a = \mathtt{T}$ or $b = \mathtt{T}$	$a, b \in \mathcal{D}_o$
$\mathfrak{L}_\wedge(c)$	$c \in \mathcal{D}_{o\to o\to o}$	$c(a)(b) = \mathtt{T}$	iff	$a = \mathtt{T}$ and $b = \mathtt{T}$	$a, b \in \mathcal{D}_o$
$\mathfrak{L}_{\Pi^\alpha}(\pi)$	$\pi \in \mathcal{D}_{(\alpha\to o)\to o}$	$\pi(f) = \mathtt{T}$	iff	$\forall a \in \mathcal{D}_\alpha\ f(a) = \mathtt{T}$	$f \in \mathcal{D}_{\alpha\to o}$
$\mathfrak{L}_{\Sigma^\alpha}(\sigma)$	$\sigma \in \mathcal{D}_{(\alpha\to o)\to o}$	$\sigma(f) = \mathtt{T}$	iff	$\exists a \in \mathcal{D}_\alpha\ f(a) = \mathtt{T}$	$f \in \mathcal{D}_{\alpha\to o}$
$\mathfrak{L}_{=^\alpha}(q)$	$q \in \mathcal{D}_{\alpha\to\alpha\to o}$	$q(a)(b) = \mathtt{T}$	iff	$a = b$	$a, b \in \mathcal{D}_\alpha$

The frame $\mathcal{D}$ realizes c if there is some a such that $\mathfrak{L}_c(a)$ holds.

Surjective Cantor Theorem: II

Suppose $\mathcal{D}$ is a

- Combinatory Frame
- $\mathcal{D}_o = \{\mathtt{T},\mathtt{F}\}$
- $\mathcal{D}$ realizes $\neg$

Then

there is no surjection in $\mathcal{D}_{\iota\to\iota\to o}$.

Surjective Cantor Theorem: Proof

$\mathcal{D}$ Combinatory, $\mathcal{D}_o = \{\mathtt{T},\mathtt{F}\}$, realizes $\neg$

Suppose $G \in \mathcal{D}_{\iota\to\iota\to o}$ Surjective

- Let Σ be signature $\{\neg\}$
- Let $\mathcal{I} : (\Sigma \cup \mathcal{P}) \longrightarrow \mathcal{D}$ such that $\mathfrak{L}_\neg(\mathcal{I}(\neg))$
- $\mathcal{D}$ combinatory $\Rightarrow \exists\mathcal{E}$ evaluating terms extending $\mathcal{I}$
- Let φ such that $\varphi(g) := G$ (g variable)
- Let $D \in \mathcal{D}_{\iota\to o}$ be $\mathcal{E}_\varphi(\lambda x_\iota [\neg [g\, x\, x]])$
- G surjective $\Rightarrow \exists X \in \mathcal{D}_\iota$ s.t. $G(X) = D$

Surjective Cantor Theorem: Proof

$\mathcal{D}$ Combinatory, $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$, realizes $\neg$

Suppose $\mathsf{G} \in \mathcal{D}_{\iota\to\iota\to o}$ Surjective

- Let $\mathsf{D} \in \mathcal{D}_{\iota\to o}$ be $\mathcal{E}_\varphi(\lambda \mathsf{x}_\iota [\neg[\mathsf{g}\,\mathsf{x}\,\mathsf{x}]])$
- G surjective $\Rightarrow \exists \mathsf{X} \in \mathcal{D}_\iota$ s.t. $\mathsf{G}(\mathsf{X}) = \mathsf{D}$

$$\begin{array}{lll} \mathsf{G}(\mathsf{X})(\mathsf{X}) = \mathtt{T} & \text{iff} & \mathsf{D}(\mathsf{X}) = \mathtt{T} \\ & \text{iff} & \mathcal{E}_\varphi(\lambda \mathsf{x}_\iota [\neg[\mathsf{g}\,\mathsf{x}\,\mathsf{x}]])(\mathsf{X}) = \mathtt{T} \\ & \text{iff} & \mathcal{E}_{\varphi,[\mathsf{X}/\mathsf{x}]}([\neg[\mathsf{g}\,\mathsf{x}\,\mathsf{x}]]) = \mathtt{T} \\ & \text{iff} & \mathcal{I}(\neg)(\mathsf{G}(\mathsf{X})(\mathsf{X})) = \mathtt{T} \\ & \text{iff} & \mathsf{G}(\mathsf{X})(\mathsf{X}) = \mathtt{F} \end{array}$$

$\boxed{\text{Contradiction!}}$

Combinatory Frame Without Negation

(Ignore type ι)
Define $\mathcal{D}_o := \{\mathtt{F}, \mathtt{T}\}$.
Order $\mathcal{D}_o$ by $\mathtt{F} <^o \mathtt{T}$.
Extend to function types:
Assume $\mathcal{D}_\alpha$ is defined and ordered by $\leq^\alpha$.
Assume $\mathcal{D}_\beta$ is defined and ordered by $\leq^\beta$.

Define $\mathcal{D}_{\alpha\to\beta}$ to be all monotone functions $\mathsf{f} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$.

$$\forall \mathsf{x}, \mathsf{y} \in \mathcal{D}_\alpha \centerdot \mathsf{x} \leq^\alpha \mathsf{y} \Rightarrow \mathsf{f}(\mathsf{x}) \leq^\beta \mathsf{f}(\mathsf{y})$$

Combinatory Frame Without Negation

Order $\mathcal{D}_{\alpha\to\beta}$ by $\leq^{\alpha\to\beta}$: $\mathsf{f} \leq^{\alpha\to\beta} \mathsf{g}$ iff

$$\forall \mathsf{x}, \mathsf{y} \in \mathcal{D}_\alpha \centerdot \mathsf{x} \leq^\alpha \mathsf{y} \Rightarrow \mathsf{f}(\mathsf{x}) \leq^\beta \mathsf{g}(\mathsf{y}).$$

The frame $\mathcal{D}$ is combinatory.

$\neg : \mathcal{D}_o \longrightarrow \mathcal{D}_o$ is not monotone since $\mathtt{F} \leq^o \mathtt{T}$, but $\neg(\mathtt{F}) \not\leq^o \neg(\mathtt{T})$

Hence $\mathcal{D}_{o\to o} = \{\mathsf{K}_\mathtt{F}, \mathsf{K}_\mathtt{T}, \mathsf{id}\}$.

Constructing Nonstandard Frames

(Binary) Logical Relation Frames

Start with $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota$ nonempty,

$\mathcal{R}_o \subseteq (\mathcal{D}_o)^2$ (e.g., $\leq^o$) and $\mathcal{R}_\iota \subseteq (\mathcal{D}_\iota)^2$.

Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ are reflexive binary relations.
Extend to function types:

Assume $\mathcal{D}_\alpha$, $\mathcal{R}_\alpha \subseteq (\mathcal{D}_\alpha)^2$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq (\mathcal{D}_\beta)^2$ are defined.

Define $\mathcal{D}_{\alpha\to\beta}$:

$$\{\mathsf{f} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \langle \mathsf{y}^0, \mathsf{y}^1 \rangle \in \mathcal{R}_\alpha \Rightarrow \langle \mathsf{f}(\mathsf{y}^0), \mathsf{f}(\mathsf{y}^1) \rangle \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} \subseteq (\mathcal{D}_{\alpha\to\beta})^2$:

$$\{\langle \mathsf{f}^0, \mathsf{f}^1 \rangle \mid \forall \langle \mathsf{y}^0, \mathsf{y}^1 \rangle \in \mathcal{R}_\alpha \Rightarrow \langle \mathsf{f}^0(\mathsf{y}^0), \mathsf{f}^1(\mathsf{y}^1) \rangle \in \mathcal{R}_\beta\}$$

Logical Relation Frames

$\ldots$ $\ldots$

$\mathcal{D}'$s $\mathcal{R}'$s

$\mathcal{D}_{o\to o}$ $\mathcal{D}_{\iota\to o}$ $\mathcal{D}_{o\to\iota}$ $\mathcal{D}_{\iota\to\iota}$ $\mathcal{R}_{o\to o}$ $\mathcal{R}_{\iota\to o}$ $\mathcal{R}_{o\to\iota}$ $\mathcal{R}_{\iota\to\iota}$

$\mathcal{D}_o$ $\mathcal{D}_\iota$ $\mathcal{R}_o$ $\mathcal{R}_\iota$

Given: $\mathcal{D}_o$, $\mathcal{D}_\iota$, $\mathcal{R}_o$, $\mathcal{R}_\iota$

Logical Relation Frames

$\ldots$ $\ldots$

$\mathcal{D}'$s $\mathcal{R}'$s

$\mathcal{D}_{o\to o}$ $\mathcal{D}_{\iota\to o}$ $\mathcal{D}_{o\to\iota}$ $\mathcal{D}_{\iota\to\iota}$ $\mathcal{R}_{o\to o}$ $\mathcal{R}_{\iota\to o}$ $\mathcal{R}_{o\to\iota}$ $\mathcal{R}_{\iota\to\iota}$

$\mathcal{D}_o$ $\mathcal{D}_\iota$ $\mathcal{R}_o$ $\mathcal{R}_\iota$

Define: $\mathcal{D}_{\alpha\to\beta}$, $\alpha, \beta \in \{o, \iota\}$

Logical Relation Frames

$\ldots$ $\ldots$

$\mathcal{D}'$s $\mathcal{R}'$s

$\mathcal{D}_{o\to o}$ $\mathcal{D}_{\iota\to o}$ $\mathcal{D}_{o\to\iota}$ $\mathcal{D}_{\iota\to\iota}$ $\mathcal{R}_{o\to o}$ $\mathcal{R}_{\iota\to o}$ $\mathcal{R}_{o\to\iota}$ $\mathcal{R}_{\iota\to\iota}$

$\mathcal{D}_o$ $\mathcal{D}_\iota$ $\mathcal{R}_o$ $\mathcal{R}_\iota$

Define: $\mathcal{R}_{\alpha\to\beta}$, $\alpha, \beta \in \{o, \iota\}$

Logical Relation Frames

. . . . . .

$\mathcal{D}'$s $\mathcal{R}'$s

$\mathcal{D}_{o\to o}$ $\mathcal{D}_{\iota\to o}$ $\mathcal{D}_{o\to\iota}$ $\mathcal{D}_{\iota\to\iota}$ $\mathcal{R}_{o\to o}$ $\mathcal{R}_{\iota\to o}$ $\mathcal{R}_{o\to\iota}$ $\mathcal{R}_{\iota\to\iota}$

$\mathcal{D}_o$ $\mathcal{D}_\iota$ $\mathcal{R}_o$ $\mathcal{R}_\iota$

Define: Next $\mathcal{D}'$s

Logical Relation Frames

. . . . . .

$\mathcal{D}'$s $\mathcal{R}'$s

$\mathcal{D}_{o\to o}$ $\mathcal{D}_{\iota\to o}$ $\mathcal{D}_{o\to\iota}$ $\mathcal{D}_{\iota\to\iota}$ $\mathcal{R}_{o\to o}$ $\mathcal{R}_{\iota\to o}$ $\mathcal{R}_{o\to\iota}$ $\mathcal{R}_{\iota\to\iota}$

$\mathcal{D}_o$ $\mathcal{D}_\iota$ $\mathcal{R}_o$ $\mathcal{R}_\iota$

Define: Next $\mathcal{R}'$s

Logical Relation Frames

. . . . . .

$\mathcal{D}'$s $\mathcal{R}'$s

$\mathcal{D}_{o\to o}$ $\mathcal{D}_{\iota\to o}$ $\mathcal{D}_{o\to\iota}$ $\mathcal{D}_{\iota\to\iota}$ $\mathcal{R}_{o\to o}$ $\mathcal{R}_{\iota\to o}$ $\mathcal{R}_{o\to\iota}$ $\mathcal{R}_{\iota\to\iota}$

$\mathcal{D}_o$ $\mathcal{D}_\iota$ $\mathcal{R}_o$ $\mathcal{R}_\iota$

$\cdots$ *and so on*

Logical Relation Frames

Fact 1: Each $\mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^2$ is reflexive.

Fact 2: $\mathcal{D}$ is a combinatory frame.

No Proofs Yet...Wait...

Logical Relation Frames

Construction works with relations of any arity.
Let A be a nonempty set.

function $p : A \longrightarrow \mathcal{D}_\alpha \quad \sim \quad$ A-tuple $\langle p(i)\rangle_{i\in A}$

of elements of $\mathcal{D}_\alpha$

set of functions $\mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^A \quad \sim \quad$ A-ary relation on $\mathcal{D}_\alpha$

Logical Relation Frames

Let $x \in \mathcal{D}_\alpha$ and $K_x : A \longrightarrow \mathcal{D}_\alpha$ be the

constant function $K_x(i) = x$ for all $i \in A$.

constant function $K_x \quad \sim \quad$ A-tuple $\langle x\rangle_{i\in A}$.

Instead of *reflexivity* of $\mathcal{R}_\alpha$,

we will need all constant functions K_x to be in $\mathcal{R}_\alpha$.

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.
Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}, \mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.
Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{(p^0, p^1)} \in \mathcal{R}_\alpha \quad \boxed{\langle f(p^0), f(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{(q^0, q^1)} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{(p^0, p^1)} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.
Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}, \mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.
Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{(p^0, p^1)} \in \mathcal{R}_\alpha \quad \boxed{\langle f(p^0), f(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{(q^0, q^1)} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{(p^0, p^1)} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle f(p^0), f(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle f(p^0), f(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle f(p^0), f(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle f(p(0)), f(p(1))\rangle} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o$, $\mathcal{D}_\iota$, $\mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha$, $\mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{i \mapsto f(p(i))} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o$, $\mathcal{D}_\iota$, $\mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha$, $\mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{(f \circ p)} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o$, $\mathcal{D}_\iota$, $\mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha$, $\mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{(f \circ p)} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o$, $\mathcal{D}_\iota$, $\mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha$, $\mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{(f \circ p)} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{\langle q^0(p^0), q^1(p^1)\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{\mathsf{f} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{\mathsf{p}} \in \mathcal{R}_\alpha \quad \boxed{(\mathsf{f} \circ \mathsf{p})} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{\mathsf{q}} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{\mathsf{p}} \in \mathcal{R}_\alpha \quad \boxed{\langle \mathsf{q}(0)(\mathsf{p}(0)), \mathsf{q}(1)(\mathsf{p}(1))\rangle} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{\mathsf{f} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{\mathsf{p}} \in \mathcal{R}_\alpha \quad \boxed{(\mathsf{f} \circ \mathsf{p})} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{\mathsf{q}} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{\mathsf{p}} \in \mathcal{R}_\alpha \quad \boxed{\mathsf{i} \mapsto \mathsf{q}(\mathsf{i})(\mathsf{p}(\mathsf{i}))} \in \mathcal{R}_\beta\}$$

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{\mathsf{f} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{\mathsf{p}} \in \mathcal{R}_\alpha \quad \boxed{(\mathsf{f} \circ \mathsf{p})} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{\mathsf{q}} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{\mathsf{p}} \in \mathcal{R}_\alpha \quad \boxed{\mathsf{S}(\mathsf{q}, \mathsf{p})} \in \mathcal{R}_\beta\}$$

where $\mathsf{S}(\mathsf{q}, \mathsf{p})(\mathsf{i}) := \mathsf{q}(\mathsf{i})(\mathsf{p}(\mathsf{i}))$.

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{\mathsf{f} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{\mathsf{p}} \in \mathcal{R}_\alpha \quad \boxed{(\mathsf{f} \circ \mathsf{p})} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{\mathsf{q}} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{\mathsf{p}} \in \mathcal{R}_\alpha \quad \boxed{\mathsf{S}(\mathsf{q}, \mathsf{p})} \in \mathcal{R}_\beta\}$$

where $\mathsf{S}(\mathsf{q}, \mathsf{p})(\mathsf{i}) := \mathsf{q}(\mathsf{i})(\mathsf{p}(\mathsf{i}))$.

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{2}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{2}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{2}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{2}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{(f \circ p)} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{2}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{S(q,p)} \in \mathcal{R}_\beta\}$$

where $S(q,p)(i) := q(i)(p(i))$.

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{A}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{A}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{A}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{A}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{(f \circ p)} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{A}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{S(q,p)} \in \mathcal{R}_\beta\}$$

where $S(q,p)(i) := q(i)(p(i))$.

Generalizing Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^{\boxed{A}}$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^{\boxed{A}}$.

Extend to function types:

Assume $\mathcal{D}_\alpha, \mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha^{\boxed{A}}$, $\mathcal{D}_\beta$ and $\mathcal{R}_\beta \subseteq \mathcal{D}_\beta^{\boxed{A}}$ are defined.

Define $\mathcal{D}_{\alpha\to\beta} :=$

$$\{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{(f \circ p)} \in \mathcal{R}_\beta\}$$

Define $\mathcal{R}_{\alpha\to\beta} :=$

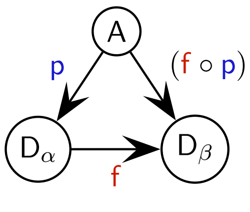
$$\{\boxed{q} \in (\mathcal{D}_{\alpha\to\beta})^{\boxed{A}} \mid \forall \boxed{p} \in \mathcal{R}_\alpha \quad \boxed{S(q,p)} \in \mathcal{R}_\beta\}$$

where $S(q,p)(i) := q(i)(p(i))$.

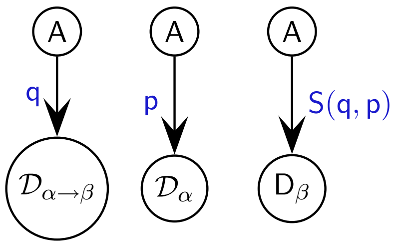
Logical Relation Frames

Start with $\mathcal{D}_o, \mathcal{D}_\iota, \mathcal{R}_o \subseteq \mathcal{D}_o^A$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^A$.

$f \in \mathcal{D}_{\alpha\to\beta}$ iff $\forall p \in \mathcal{R}_\alpha (f \circ p) \in \mathcal{R}_\beta$



$q \in \mathcal{R}_{\alpha\to\beta}$ iff $\forall p \in \mathcal{R}_\alpha S(q,p) \in \mathcal{R}_\beta$



Logical Relation Frames

Defn: This $\mathcal{D}$ is the A-ary logical relation extension of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$ (or, A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$).

Defn: If $\mathcal{D}$ is a frame, $\mathcal{D}$ is an A-ary logical relation frame with relation $\mathcal{R}$.

Facts:

- If $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions, then each $\mathcal{R}_\alpha \subseteq (\mathcal{D}_\alpha)^{\mathsf{A}}$ contains all constant functions.
- If $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty and $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions, then $\mathcal{D}$ is a combinatory frame.

Proofs...

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.

Proof: Case Split.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.
Assume α is a function type $\gamma \to \beta$.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.
Assume α is a function type $\gamma \rightarrow \beta$.
Show: $\mathcal{R}_{\gamma\rightarrow\beta}$ contains all constant functions.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.
Assume α is a function type $\gamma \rightarrow \beta$.
Show: $\mathcal{R}_{\gamma\rightarrow\beta}$ contains all constant functions.
Let $\mathsf{f} \in \mathcal{D}_{\gamma\rightarrow\beta}$ be given.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.
Assume α is a function type $\gamma \rightarrow \beta$.
Show: $\mathcal{R}_{\gamma\rightarrow\beta}$ contains all constant functions.
Let $\mathsf{f} \in \mathcal{D}_{\gamma\rightarrow\beta}$ be given.
Show: $\mathsf{K}_{\mathsf{f}} \in \mathcal{R}_{\gamma\rightarrow\beta}$.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.
Assume α is a function type $\gamma \rightarrow \beta$.
Show: $\mathcal{R}_{\gamma\rightarrow\beta}$ contains all constant functions.
Let $\mathsf{f} \in \mathcal{D}_{\gamma\rightarrow\beta}$ be given.
Show: $\mathsf{K}_{\mathsf{f}} \in \mathcal{R}_{\gamma\rightarrow\beta}$.
I.e., show: for all $\mathsf{p} \in \mathcal{R}_\gamma$, $\mathsf{S}(\mathsf{K}_{\mathsf{f}}, \mathsf{p}) \in \mathcal{R}_\beta$.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.
Assume α is a function type $\gamma \to \beta$.
Show: $\mathcal{R}_{\gamma \to \beta}$ contains all constant functions.
Let $f \in \mathcal{D}_{\gamma \to \beta}$ be given.
Show: $K_f \in \mathcal{R}_{\gamma \to \beta}$.
I.e., show: for all $p \in \mathcal{R}_\gamma$, $S(K_f, p) \in \mathcal{R}_\beta$.
Let $p \in \mathcal{R}_\gamma$ be given.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.
Assume α is a function type $\gamma \to \beta$.
Show: $\mathcal{R}_{\gamma \to \beta}$ contains all constant functions.
Let $f \in \mathcal{D}_{\gamma \to \beta}$ be given.
Show: $K_f \in \mathcal{R}_{\gamma \to \beta}$.
I.e., show: for all $p \in \mathcal{R}_\gamma$, $S(K_f, p) \in \mathcal{R}_\beta$.
Let $p \in \mathcal{R}_\gamma$ be given.
For $i \in A$, $S(K_f, p)(i) = K_f(i)(p(i)) = f(p(i))$.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.
Assume α is a function type $\gamma \to \beta$.
Show: $\mathcal{R}_{\gamma \to \beta}$ contains all constant functions.
Let $f \in \mathcal{D}_{\gamma \to \beta}$ be given.
Show: $K_f \in \mathcal{R}_{\gamma \to \beta}$.
I.e., show: for all $p \in \mathcal{R}_\gamma$, $S(K_f, p) \in \mathcal{R}_\beta$.
Let $p \in \mathcal{R}_\gamma$ be given.
For $i \in A$, $S(K_f, p)(i) = K_f(i)(p(i)) = f(p(i))$.
Hence $S(K_f, p) = (f \circ p)$.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{R}_\alpha$ contains all constant functions.
Proof: Case Split. For $\alpha \in \{o, \iota\}$ assertion holds by assumption.
Assume α is a function type $\gamma \to \beta$.
Show: $\mathcal{R}_{\gamma \to \beta}$ contains all constant functions.
Let $f \in \mathcal{D}_{\gamma \to \beta}$ be given.
Show: $K_f \in \mathcal{R}_{\gamma \to \beta}$.
I.e., show: for all $p \in \mathcal{R}_\gamma$, $S(K_f, p) \in \mathcal{R}_\beta$.
Let $p \in \mathcal{R}_\gamma$ be given.
For $i \in A$, $S(K_f, p)(i) = K_f(i)(p(i)) = f(p(i))$.
Hence $S(K_f, p) = (f \circ p)$.

Note $(f \circ p) \in \mathcal{R}_\beta$ since $f \in \mathcal{D}_{\gamma \to \beta}$. QED

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{D}_\alpha$ is nonempty.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{D}_\alpha$ is nonempty.
Proof: Induction on types. Base cases by assumption.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{D}_\alpha$ is nonempty.
Proof: Induction on types. Base cases by assumption.
Assume $\mathcal{D}_\beta$ is nonempty. Choose some $b \in \mathcal{D}_\beta$.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{D}_\alpha$ is nonempty.
Proof: Induction on types. Base cases by assumption.
Assume $\mathcal{D}_\beta$ is nonempty. Choose some $b \in \mathcal{D}_\beta$.
Let $f : \mathcal{D}_\gamma \longrightarrow \mathcal{D}_\beta$ be $f(x) := b$ for all $x \in \mathcal{D}_\gamma$.

Logical Relation Frames

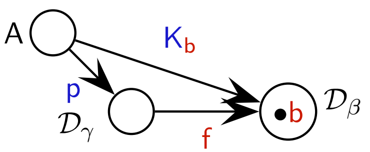
Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{D}_\alpha$ is nonempty.
Proof: Induction on types. Base cases by assumption.
Assume $\mathcal{D}_\beta$ is nonempty. Choose some $b \in \mathcal{D}_\beta$.
Let $f : \mathcal{D}_\gamma \longrightarrow \mathcal{D}_\beta$ be $f(x) := b$ for all $x \in \mathcal{D}_\gamma$.
Claim: $f \in \mathcal{D}_{\gamma\to\beta}$.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{D}_\alpha$ is nonempty.
Proof: Induction on types. Base cases by assumption.
Assume $\mathcal{D}_\beta$ is nonempty. Choose some $b \in \mathcal{D}_\beta$.
Let $f : \mathcal{D}_\gamma \longrightarrow \mathcal{D}_\beta$ be $f(x) := b$ for all $x \in \mathcal{D}_\gamma$.
Claim: $f \in \mathcal{D}_{\gamma\to\beta}$.
Let $p \in \mathcal{R}_\gamma$ be given. Check: $(f \circ p) \in \mathcal{R}_\beta$.

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{D}_\alpha$ is nonempty.
Proof: Induction on types. Base cases by assumption.
Assume $\mathcal{D}_\beta$ is nonempty. Choose some $b \in \mathcal{D}_\beta$.
Let $f : \mathcal{D}_\gamma \longrightarrow \mathcal{D}_\beta$ be $f(x) := b$ for all $x \in \mathcal{D}_\gamma$.
Claim: $f \in \mathcal{D}_{\gamma\to\beta}$.
Let $p \in \mathcal{R}_\gamma$ be given. Check: $(f \circ p) \in \mathcal{R}_\beta$.
Note: $(f \circ p) = K_b$ (constant b function with domain A).



Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{D}_\alpha$ is nonempty.
Proof: Induction on types. Base cases by assumption.
Assume $\mathcal{D}_\beta$ is nonempty. Choose some $b \in \mathcal{D}_\beta$.
Let $f : \mathcal{D}_\gamma \longrightarrow \mathcal{D}_\beta$ be $f(x) := b$ for all $x \in \mathcal{D}_\gamma$.
Claim: $f \in \mathcal{D}_{\gamma\to\beta}$.
Let $p \in \mathcal{R}_\gamma$ be given. Check: $(f \circ p) \in \mathcal{R}_\beta$.
Note: $(f \circ p) = K_b$ (constant b function with domain A).
By previous lemma, $(f \circ p) \in \mathcal{R}_\beta$. QED

Logical Relation Frames

Lemma: Let $\mathcal{D}$ be the A-ary l.r.e. of $\langle \mathcal{D}_\iota, \mathcal{D}_o, \mathcal{R}_\iota, \mathcal{R}_o \rangle$.
Assume $\mathcal{D}_o$ and $\mathcal{D}_\iota$ are nonempty.
Assume $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.
Then each $\mathcal{D}$ is a combinatory frame.
Proof: Know $\mathcal{D}$ is a frame:
Each $\mathcal{D}_\alpha$ is nonempty.
Each $\mathcal{D}_{\alpha\to\beta} \subseteq \mathcal{D}_\beta^{\mathcal{D}_\alpha}$.

Must Check Combinatory: $\mathtt{k}$ and $\mathtt{s}$

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \,\forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \,\forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.
Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.
Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \,\forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.
Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.
Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.
First check: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \; \forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.

Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.

Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.

First check: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.

Let $a \in \mathcal{D}_\alpha$ be given. Check: $\mathtt{k}(a) \in \mathcal{D}_{\beta\to\alpha}$.

UNIVERSITÄT DES SAARLANDES

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \; \forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.

Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.

Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.

First check: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.

Let $a \in \mathcal{D}_\alpha$ be given. Check: $\mathtt{k}(a) \in \mathcal{D}_{\beta\to\alpha}$.

Let $p \in \mathcal{R}_\beta$ be given.

©Benzmüller & Brown, 2006

UNIVERSITÄT DES SAARLANDES

SEMHOL'06-[1] – p.63

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \; \forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.

Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.

Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.

First check: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.

Let $a \in \mathcal{D}_\alpha$ be given. Check: $\mathtt{k}(a) \in \mathcal{D}_{\beta\to\alpha}$.

Let $p \in \mathcal{R}_\beta$ be given.

$(\mathtt{k}(a) \circ p) : A \longrightarrow \mathcal{D}_\alpha$ is constant a function.

Hence: $(\mathtt{k}(a) \circ p) \in \mathcal{R}_\alpha$.

©Benzmüller & Brown, 2006

UNIVERSITÄT DES SAARLANDES

SEMHOL'06-[1] – p.63

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \; \forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.

Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.

Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.

First check: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.

Let $a \in \mathcal{D}_\alpha$ be given. Check: $\mathtt{k}(a) \in \mathcal{D}_{\beta\to\alpha}$.

Let $p \in \mathcal{R}_\beta$ be given.

$(\mathtt{k}(a) \circ p) : A \longrightarrow \mathcal{D}_\alpha$ is constant a function.

Hence: $(\mathtt{k}(a) \circ p) \in \mathcal{R}_\alpha$.

Hence: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.

©Benzmüller & Brown, 2006

UNIVERSITÄT DES SAARLANDES

SEMHOL'06-[1] – p.63

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \,\forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.
Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.
Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.
Know: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.
To check $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$, let $q \in \mathcal{R}_\alpha$ be given.

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \,\forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.
Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.
Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.
Know: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.
To check $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$, let $q \in \mathcal{R}_\alpha$ be given.
In order to check $(\mathtt{k} \circ q) \in \mathcal{R}_{\beta\to\alpha}$, let $p \in \mathcal{R}_\beta$ be given.

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \,\forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.
Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.
Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.
Know: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.
To check $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$, let $q \in \mathcal{R}_\alpha$ be given.
In order to check $(\mathtt{k} \circ q) \in \mathcal{R}_{\beta\to\alpha}$, let $p \in \mathcal{R}_\beta$ be given.
Now, we must check that $S(\mathtt{k} \circ q, p) \in \mathcal{R}_\alpha$.

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(a)(b) = a \,\forall a \in \mathcal{D}_\alpha, b \in \mathcal{D}_\beta$.
Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(a)(b) := a$.
Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.
Know: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.
To check $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$, let $q \in \mathcal{R}_\alpha$ be given.
In order to check $(\mathtt{k} \circ q) \in \mathcal{R}_{\beta\to\alpha}$, let $p \in \mathcal{R}_\beta$ be given.
Now, we must check that $S(\mathtt{k} \circ q, p) \in \mathcal{R}_\alpha$.
For each $i \in A$, we compute

$$S(\mathtt{k} \circ q, p)(i) = (\mathtt{k} \circ q)(i)(p(i)) = \mathtt{k}(q(i))(p(i)) = q(i).$$

Logical Relation Frames: K

Check: $\exists \mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha} \ni \mathtt{k}(\mathtt{a})(\mathtt{b}) = \mathtt{a}\ \forall \mathtt{a} \in \mathcal{D}_\alpha, \mathtt{b} \in \mathcal{D}_\beta$.
Define $\mathtt{k}$ as function $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\alpha^{\mathcal{D}_\beta}$ by $\mathtt{k}(\mathtt{a})(\mathtt{b}) := \mathtt{a}$.
Check: $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$.
Know: $\mathtt{k} : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_{\beta\to\alpha}$.
To check $\mathtt{k} \in \mathcal{D}_{\alpha\to\beta\to\alpha}$, let $\mathsf{q} \in \mathcal{R}_\alpha$ be given.
In order to check $(\mathtt{k} \circ \mathsf{q}) \in \mathcal{R}_{\beta\to\alpha}$, let $\mathsf{p} \in \mathcal{R}_\beta$ be given.
Now, we must check that $\mathsf{S}(\mathtt{k} \circ \mathsf{q}, \mathsf{p}) \in \mathcal{R}_\alpha$.
For each $\mathsf{i} \in \mathsf{A}$, we compute

$$\mathsf{S}(\mathtt{k} \circ \mathsf{q}, \mathsf{p})(\mathsf{i}) = (\mathtt{k} \circ \mathsf{q})(\mathsf{i})(\mathsf{p}(\mathsf{i})) = \mathtt{k}(\mathsf{q}(\mathsf{i}))(\mathsf{p}(\mathsf{i})) = \mathsf{q}(\mathsf{i}).$$

Thus $\mathsf{S}(\mathtt{k} \circ \mathsf{q}, \mathsf{p}) = \mathsf{q} \in \mathcal{R}_\alpha$.

Logical Relation Frames: S

Define $\mathtt{s}(\mathtt{g})(\mathtt{f})(\mathtt{a}) := \mathtt{g}(\mathtt{a})(\mathtt{f}(\mathtt{a}))$
Check: $\mathtt{s}(\mathtt{g})(\mathtt{f}) \in \mathcal{D}_*$

$$(\mathtt{s}(\mathtt{g})(\mathtt{f}) \circ \mathsf{p}) = \mathsf{S}(\mathtt{g} \circ \mathsf{p}, \mathtt{f} \circ \mathsf{p}) \in \mathcal{R}_*$$

Check: $\mathtt{s}(\mathtt{g}) \in \mathcal{D}_*$

$$\mathsf{S}(\mathtt{s}(\mathtt{g}) \circ \mathsf{q}, \mathsf{p}) = \mathsf{S}(\mathtt{g} \circ \mathsf{p}, \mathsf{S}(\mathsf{q}, \mathsf{p})) \in \mathcal{R}_*$$

Check: $\mathtt{s} \in \mathcal{D}_*$

$$\mathsf{S}(\mathsf{S}(\mathtt{s} \circ \mathsf{r}, \mathsf{q}), \mathsf{p}) = \mathsf{S}(\mathsf{S}(\mathsf{r}, \mathsf{q}), \mathsf{p}) \in \mathcal{R}_*$$

Review

- Given A, $\mathcal{D}_o$, $\mathcal{D}_\iota$ nonempty and $\mathcal{R}_o \subseteq \mathcal{D}_o^A$ and $\mathcal{R}_\iota \subseteq \mathcal{D}_\iota^A$ containing all constant functions
- Define $\mathcal{D}_\alpha$ and $\mathcal{R}_\alpha \subseteq \mathcal{D}_\alpha$
- $\mathcal{D}$ combinatory
- $\mathcal{D}$ evaluates λ-terms
- $\mathcal{D}$ may or may not realize logical constants

Frames with Specified Sets

Let A be nonempty and $\mathcal{B} \subseteq \mathcal{P}A$. Assume $\emptyset \in \mathcal{B}$ and $A \in \mathcal{B}$.

Define $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$ and $\mathcal{D}_\iota := A$.

Define $\mathcal{R}_o \subseteq (\mathcal{D}_o)^A$: $\{\mathsf{p} : A \longrightarrow \mathcal{D}_o \mid \mathsf{p}^{-1}(\mathtt{T}) \in \mathcal{B}\} = \{\chi_X \mid X \in \mathcal{B}\}$

Define $\mathcal{R}_\iota \subseteq (\mathcal{D}_\iota)^A$: $\{\mathsf{p} : A \longrightarrow \mathcal{D}_\iota \mid \mathsf{p}^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

Fact: $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.

Proof:

Frames with Specified Sets

Let A be nonempty and $\mathcal{B} \subseteq \mathcal{P}A$. Assume $\emptyset \in \mathcal{B}$ and $A \in \mathcal{B}$.

Define $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$ and $\mathcal{D}_\iota := A$.

Define $\mathcal{R}_o \subseteq (\mathcal{D}_o)^A$: $\{p : A \longrightarrow \mathcal{D}_o \mid p^{-1}(\mathtt{T}) \in \mathcal{B}\} = \{\chi_X \mid X \in \mathcal{B}\}$

Define $\mathcal{R}_\iota \subseteq (\mathcal{D}_\iota)^A$: $\{p : A \longrightarrow \mathcal{D}_\iota \mid p^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

Fact: $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.

Proof:

$$K_{\mathtt{T}} = \chi_A \qquad (K_{\mathtt{T}})^{-1}(\mathtt{T}) = A \in \mathcal{B}$$

$$K_{\mathtt{F}} = \chi_\emptyset \qquad (K_{\mathtt{F}})^{-1}(\mathtt{T}) = \emptyset \in \mathcal{B}$$

Hence $K_{\mathtt{T}}, K_{\mathtt{F}} \in \mathcal{R}_o$.

UNIVERSITÄT DES SAARLANDES

Frames with Specified Sets

Let A be nonempty and $\mathcal{B} \subseteq \mathcal{P}A$. Assume $\emptyset \in \mathcal{B}$ and $A \in \mathcal{B}$.

Define $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$ and $\mathcal{D}_\iota := A$.

Define $\mathcal{R}_o \subseteq (\mathcal{D}_o)^A$: $\{p : A \longrightarrow \mathcal{D}_o \mid p^{-1}(\mathtt{T}) \in \mathcal{B}\} = \{\chi_X \mid X \in \mathcal{B}\}$

Define $\mathcal{R}_\iota \subseteq (\mathcal{D}_\iota)^A$: $\{p : A \longrightarrow \mathcal{D}_\iota \mid p^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

Fact: $\mathcal{R}_o$ and $\mathcal{R}_\iota$ contain all constant functions.

Proof: For each $a \in A$ and $X \in \mathcal{B}$

$$\boxed{(K_a)^{-1}(X) = A \in \mathcal{B} \text{ if } a \in X} \text{ and } \boxed{(K_a)^{-1}(X) = \emptyset \in \mathcal{B} \text{ if } a \notin X}$$

Hence $K_a \in \mathcal{R}_\iota$.

© Benzmüller & Brown, 2006

UNIVERSITÄT DES SAARLANDES

SEMHOL'06-[1] – p.66

Frames with Specified Sets

Let A be nonempty and $\mathcal{B} \subseteq \mathcal{P}A$. Assume $\emptyset \in \mathcal{B}$ and $A \in \mathcal{B}$.

Define $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$ and $\mathcal{D}_\iota := A$.

Define $\mathcal{R}_o \subseteq (\mathcal{D}_o)^A$: $\{p : A \longrightarrow \mathcal{D}_o \mid p^{-1}(\mathtt{T}) \in \mathcal{B}\} = \{\chi_X \mid X \in \mathcal{B}\}$

Define $\mathcal{R}_\iota \subseteq (\mathcal{D}_\iota)^A$: $\{p : A \longrightarrow \mathcal{D}_\iota \mid p^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

Let $\mathcal{D}$ be the A-ary l.r.e. with relation $\mathcal{R}$.

By Previous Results: $\mathcal{D}$ is a combinatory frame.

© Benzmüller & Brown, 2006

UNIVERSITÄT DES SAARLANDES

SEMHOL'06-[1] – p.66

Frames with Specified Sets

Let A be nonempty and $\mathcal{B} \subseteq \mathcal{P}A$. Assume $\emptyset \in \mathcal{B}$ and $A \in \mathcal{B}$.

Define $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$ and $\mathcal{D}_\iota := A$.

Define $\mathcal{R}_o \subseteq (\mathcal{D}_o)^A$: $\{p : A \longrightarrow \mathcal{D}_o \mid p^{-1}(\mathtt{T}) \in \mathcal{B}\} = \{\chi_X \mid X \in \mathcal{B}\}$

Define $\mathcal{R}_\iota \subseteq (\mathcal{D}_\iota)^A$: $\{p : A \longrightarrow \mathcal{D}_\iota \mid p^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

Let $\mathcal{D}$ be the A-ary l.r.e. with relation $\mathcal{R}$.

By Previous Results: $\mathcal{D}$ is a combinatory frame.

We will show: $\mathcal{D}_{\iota\to o} = \{\chi_X \mid X \in \mathcal{B}\} \cong \mathcal{B}$

© Benzmüller & Brown, 2006

UNIVERSITÄT DES SAARLANDES

SEMHOL'06-[1] – p.66

Frames with Specified Sets

Let A be nonempty and $\mathcal{B} \subseteq \mathcal{P}\mathsf{A}$. Assume $\emptyset \in \mathcal{B}$ and $\mathsf{A} \in \mathcal{B}$.

Define $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$ and $\mathcal{D}_\iota := \mathsf{A}$.

Define $\mathcal{R}_o \subseteq (\mathcal{D}_o)^{\mathsf{A}}$: $\{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_o \mid \mathsf{p}^{-1}(\mathtt{T}) \in \mathcal{B}\} = \{\chi_{\mathsf{X}} \mid \mathsf{X} \in \mathcal{B}\}$

Define $\mathcal{R}_\iota \subseteq (\mathcal{D}_\iota)^{\mathsf{A}}$: $\{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_\iota \mid \mathsf{p}^{-1}(\mathsf{X}) \in \mathcal{B}$ whenever $\mathsf{X} \in \mathcal{B}\}$

Let $\mathcal{D}$ be the A-ary l.r.e. with relation $\mathcal{R}$.

By Previous Results: $\mathcal{D}$ is a combinatory frame.

We will show: $\mathcal{D}_{\iota\to o} = \{\chi_{\mathsf{X}} \mid \mathsf{X} \in \mathcal{B}\} \cong \mathcal{B}$

Idea: We *specified* $\mathcal{D}_\iota$ and $\mathcal{D}_{\iota\to o}$ by giving A and $\mathcal{B}$,

then extended definition to all types using relations.

Frames with Specified Sets

$\ldots \qquad \ldots$

$\mathcal{D}_{(o\to o)\to o} \quad \mathcal{D}_{(\iota\to o)\to o} \quad \ldots \quad \mathcal{D}_{(\iota\to\iota)\to o} \quad \mathcal{D}_{(\iota\to\iota)\to\iota}$

$\mathcal{D}_{o\to o\to o} \quad \mathcal{D}_{(\iota\to o)\to\iota} \quad \mathcal{D}_{\iota\to\iota\to o} \quad \mathcal{D}_{\iota\to\iota\to\iota}$

$\mathcal{D}_{o\to o} \quad \mathcal{D}_{\iota\to o} \cong \mathcal{B} \quad \mathcal{D}_{o\to\iota} \quad \mathcal{D}_{\iota\to\iota}$

$\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\} \qquad \mathcal{D}_\iota = \mathsf{A}$

Specified Sets: $\mathcal{D}_{\iota\to\alpha} \subseteq \mathcal{R}_\alpha$

$$\mathcal{D}_{\iota\to\alpha} \subseteq (\mathcal{D}_\alpha)^{\mathcal{D}_\iota} = (\mathcal{D}_\alpha)^{\mathsf{A}} \supseteq \mathcal{R}_\alpha.$$

$$\mathcal{R}_\iota = \{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_\iota \mid \mathsf{p}^{-1}(\mathsf{X}) \in \mathcal{B} \text{ whenever } \mathsf{X} \in \mathcal{B}\}$$

$$\mathcal{D}_{\iota\to\alpha} = \{\mathsf{f} : \mathsf{A} \longrightarrow \mathcal{D}_\alpha \mid \forall \mathsf{p} \in \mathcal{R}_\iota \quad (\mathsf{f} \circ \mathsf{p}) \in \mathcal{R}_\alpha\}$$

Specified Sets: $\mathcal{D}_{\iota\to\alpha} \subseteq \mathcal{R}_\alpha$

$$\mathcal{D}_{\iota\to\alpha} \subseteq (\mathcal{D}_\alpha)^{\mathcal{D}_\iota} = (\mathcal{D}_\alpha)^{\mathsf{A}} \supseteq \mathcal{R}_\alpha.$$

$$\mathcal{R}_\iota = \{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_\iota \mid \mathsf{p}^{-1}(\mathsf{X}) \in \mathcal{B} \text{ whenever } \mathsf{X} \in \mathcal{B}\}$$

$$\mathcal{D}_{\iota\to\alpha} = \{\mathsf{f} : \mathsf{A} \longrightarrow \mathcal{D}_\alpha \mid \forall \mathsf{p} \in \mathcal{R}_\iota \quad (\mathsf{f} \circ \mathsf{p}) \in \mathcal{R}_\alpha\}$$

Note: $\mathsf{id} \in \mathcal{R}_\iota$.

Specified Sets: $\mathcal{D}_{\iota\to\alpha} \subseteq \mathcal{R}_\alpha$

$$\mathcal{D}_{\iota\to\alpha} \subseteq (\mathcal{D}_\alpha)^{\mathcal{D}_\iota} = (\mathcal{D}_\alpha)^{\mathsf{A}} \supseteq \mathcal{R}_\alpha.$$

$$\mathcal{R}_\iota = \{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_\iota \mid \mathsf{p}^{-1}(\mathsf{X}) \in \mathcal{B} \text{ whenever } \mathsf{X} \in \mathcal{B}\}$$

$$\mathcal{D}_{\iota\to\alpha} = \{\mathsf{f} : \mathsf{A} \longrightarrow \mathcal{D}_\alpha \mid \forall \mathsf{p} \in \mathcal{R}_\iota \quad (\mathsf{f} \circ \mathsf{p}) \in \mathcal{R}_\alpha\}$$

Note: $\mathsf{id} \in \mathcal{R}_\iota$.

For any $\mathsf{f} \in \mathcal{D}_{\iota\to\alpha}$, $\mathsf{f} = (\mathsf{f} \circ \mathsf{id}) \in \mathcal{R}_\alpha$.

$$\mathcal{D}_{\iota\to\alpha} \subseteq \mathcal{R}_\alpha$$

Specified Sets: $\mathcal{D}_{\iota\to\alpha} \subseteq \mathcal{R}_\alpha$

$$\mathcal{D}_{\iota\to\alpha} \subseteq (\mathcal{D}_\alpha)^{\mathcal{D}_\iota} = (\mathcal{D}_\alpha)^{\mathsf{A}} \supseteq \mathcal{R}_\alpha.$$

$$\mathcal{R}_\iota = \{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_\iota \mid \mathsf{p}^{-1}(\mathsf{X}) \in \mathcal{B} \text{ whenever } \mathsf{X} \in \mathcal{B}\}$$

$$\mathcal{D}_{\iota\to\alpha} = \{\mathsf{f} : \mathsf{A} \longrightarrow \mathcal{D}_\alpha \mid \forall \mathsf{p} \in \mathcal{R}_\iota \quad (\mathsf{f} \circ \mathsf{p}) \in \mathcal{R}_\alpha\}$$

Note: $\mathsf{id} \in \mathcal{R}_\iota$.

For any $\mathsf{f} \in \mathcal{D}_{\iota\to\alpha}$, $\mathsf{f} = (\mathsf{f} \circ \mathsf{id}) \in \mathcal{R}_\alpha$.

$$\mathcal{D}_{\iota\to\alpha} \subseteq \mathcal{R}_\alpha$$

In particular,

$$\mathcal{D}_{\iota\to o} \subseteq \mathcal{R}_o$$

Every $\mathsf{f} \in \mathcal{D}_{\iota\to o}$ is χ_{X} for some $\mathsf{X} \in \mathcal{B}$.

Specified Sets: $\mathcal{R}_o \subseteq \mathcal{D}_{\iota\to o}$, $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$

$$\mathcal{R}_o = \{\chi_{\mathsf{X}} | \mathsf{X} \in \mathcal{B}\} = \{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_o \mid \mathsf{p}^{-1}(\mathtt{T}) \in \mathcal{B}\}$$

$$\mathcal{R}_\iota = \{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_\iota \mid \mathsf{p}^{-1}(\mathsf{X}) \in \mathcal{B} \text{ whenever } \mathsf{X} \in \mathcal{B}\}$$

$$\mathcal{D}_{\iota\to o} = \{\mathsf{f} : \mathsf{A} \longrightarrow \mathcal{D}_o \mid \forall \mathsf{p} \in \mathcal{R}_\iota \quad (\mathsf{f} \circ \mathsf{p}) \in \mathcal{R}_o\}$$

Specified Sets: $\mathcal{R}_o \subseteq \mathcal{D}_{\iota\to o}$, $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$

$$\mathcal{R}_o = \{\chi_{\mathsf{X}} | \mathsf{X} \in \mathcal{B}\} = \{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_o \mid \mathsf{p}^{-1}(\mathtt{T}) \in \mathcal{B}\}$$

$$\mathcal{R}_\iota = \{\mathsf{p} : \mathsf{A} \longrightarrow \mathcal{D}_\iota \mid \mathsf{p}^{-1}(\mathsf{X}) \in \mathcal{B} \text{ whenever } \mathsf{X} \in \mathcal{B}\}$$

$$\mathcal{D}_{\iota\to o} = \{\mathsf{f} : \mathsf{A} \longrightarrow \mathcal{D}_o \mid \forall \mathsf{p} \in \mathcal{R}_\iota \quad (\mathsf{f} \circ \mathsf{p}) \in \mathcal{R}_o\}$$

Suppose $\mathsf{X} \in \mathcal{B}$ and $\mathsf{p} \in \mathcal{R}_\iota$. $(\chi_{\mathsf{X}} \circ \mathsf{p}) \in \mathcal{R}_o$?

Specified Sets: $\mathcal{R}_o \subseteq \mathcal{D}_{\iota\to o}$, $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$

$\mathcal{R}_o = \{\chi_X | X \in \mathcal{B}\} = \{p : A \longrightarrow \mathcal{D}_o \mid p^{-1}(T) \in \mathcal{B}\}$

$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid p^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

$\mathcal{D}_{\iota\to o} = \{f : A \longrightarrow \mathcal{D}_o \mid \forall p \in \mathcal{R}_\iota \quad (f \circ p) \in \mathcal{R}_o\}$

Suppose $X \in \mathcal{B}$ and $p \in \mathcal{R}_\iota$. $(\chi_X \circ p) \in \mathcal{R}_o$?

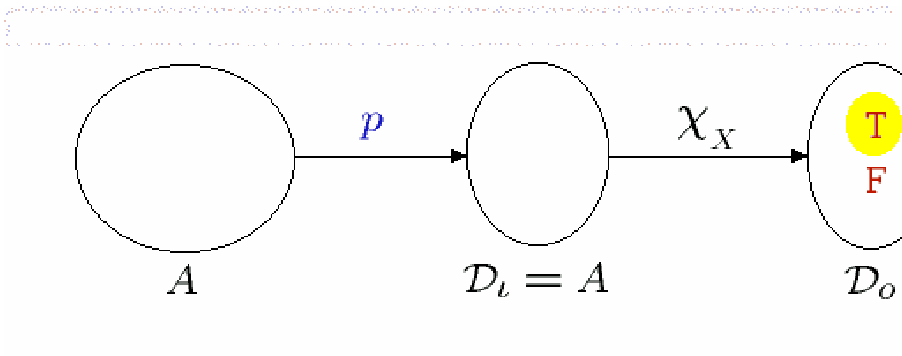
$(\chi_X \circ p)^{-1}(T) \in \mathcal{B}$?

Specified Sets: $\mathcal{R}_o \subseteq \mathcal{D}_{\iota\to o}$, $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$

$\mathcal{R}_o = \{\chi_X | X \in \mathcal{B}\} = \{p : A \longrightarrow \mathcal{D}_o \mid p^{-1}(T) \in \mathcal{B}\}$

$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid p^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

$\mathcal{D}_{\iota\to o} = \{f : A \longrightarrow \mathcal{D}_o \mid \forall p \in \mathcal{R}_\iota \quad (f \circ p) \in \mathcal{R}_o\}$

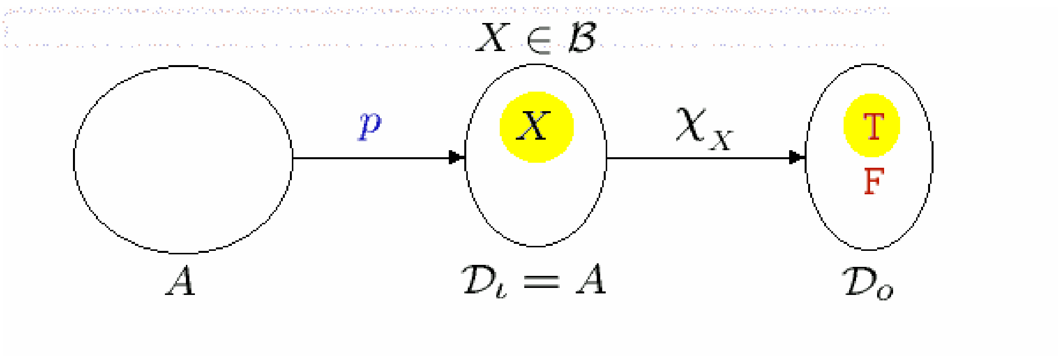


Specified Sets: $\mathcal{R}_o \subseteq \mathcal{D}_{\iota\to o}$, $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$

$\mathcal{R}_o = \{\chi_X | X \in \mathcal{B}\} = \{p : A \longrightarrow \mathcal{D}_o \mid p^{-1}(T) \in \mathcal{B}\}$

$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid p^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

$\mathcal{D}_{\iota\to o} = \{f : A \longrightarrow \mathcal{D}_o \mid \forall p \in \mathcal{R}_\iota \quad (f \circ p) \in \mathcal{R}_o\}$

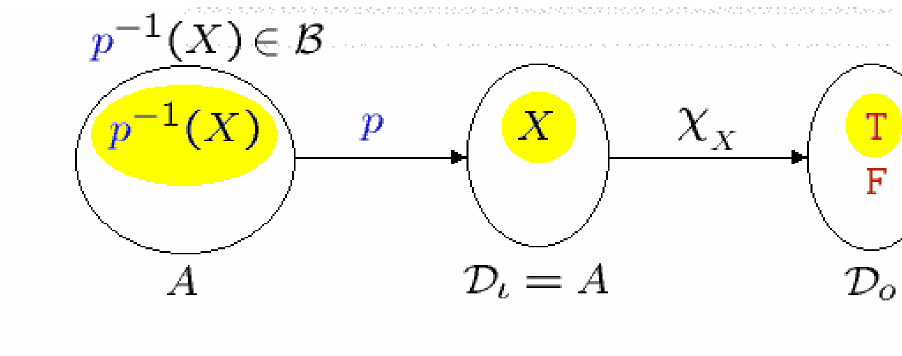


Specified Sets: $\mathcal{R}_o \subseteq \mathcal{D}_{\iota\to o}$, $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$

$\mathcal{R}_o = \{\chi_X | X \in \mathcal{B}\} = \{p : A \longrightarrow \mathcal{D}_o \mid p^{-1}(T) \in \mathcal{B}\}$

$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid p^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

$\mathcal{D}_{\iota\to o} = \{f : A \longrightarrow \mathcal{D}_o \mid \forall p \in \mathcal{R}_\iota \quad (f \circ p) \in \mathcal{R}_o\}$



Specified Sets: $\mathcal{R}_o \subseteq \mathcal{D}_{\iota\to o}$, $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$

$\mathcal{R}_o = \{\chi_X | X \in \mathcal{B}\} = \{p : A \longrightarrow \mathcal{D}_o \mid p^{-1}(T) \in \mathcal{B}\}$

$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid p^{-1}(X) \in \mathcal{B}$ whenever $X \in \mathcal{B}\}$

$\mathcal{D}_{\iota\to o} = \{f : A \longrightarrow \mathcal{D}_o \mid \forall p \in \mathcal{R}_\iota \quad (f \circ p) \in \mathcal{R}_o\}$

Suppose $X \in \mathcal{B}$ and $p \in \mathcal{R}_\iota$. $(\chi_X \circ p) \in \mathcal{R}_o$?

Yes: $(\chi_X \circ p)^{-1}(T) = p^{-1}(X) \in \mathcal{B}$ since $p \in \mathcal{R}_\iota$.

Hence $\mathcal{R}_o \subseteq \mathcal{D}_{\iota\to o}$ and $\mathcal{D}_{\iota\to o} = \mathcal{R}_o \cong \mathcal{B}$.

Specified Sets and Logical Constants

A nonempty, $\mathcal{B} \subseteq \mathcal{P}A$, $\emptyset \in \mathcal{B}$, $A \in \mathcal{B}$, $\mathcal{D}$ specified sets frame.

Let $\neg : \mathcal{D}_o \longrightarrow \mathcal{D}_o$ be the negation function. $\neg \in \mathcal{D}_{o\to o}$?

Specified Sets and Logical Constants

A nonempty, $\mathcal{B} \subseteq \mathcal{P}A$, $\emptyset \in \mathcal{B}$, $A \in \mathcal{B}$, $\mathcal{D}$ specified sets frame.

Let $\neg : \mathcal{D}_o \longrightarrow \mathcal{D}_o$ be the negation function. $\neg \in \mathcal{D}_{o\to o}$?

$\mathcal{R}_o = \{\chi_X | X \in \mathcal{B}\}$

$\mathcal{D}_{o\to o} = \{f : \mathcal{D}_o \longrightarrow \mathcal{D}_o \mid \forall p \in \mathcal{R}_o \quad (f \circ p) \in \mathcal{R}_o\}$

Specified Sets and Logical Constants

A nonempty, $\mathcal{B} \subseteq \mathcal{P}A$, $\emptyset \in \mathcal{B}$, $A \in \mathcal{B}$, $\mathcal{D}$ specified sets frame.

Let $\neg : \mathcal{D}_o \longrightarrow \mathcal{D}_o$ be the negation function. $\neg \in \mathcal{D}_{o\to o}$?

$\mathcal{R}_o = \{\chi_X | X \in \mathcal{B}\}$

$\mathcal{D}_{o\to o} = \{f : \mathcal{D}_o \longrightarrow \mathcal{D}_o \mid \forall p \in \mathcal{R}_o \quad (f \circ p) \in \mathcal{R}_o\}$

$\neg \in \mathcal{D}_{o\to o}$ iff $(\neg \circ \chi_X) \in \mathcal{R}_o$ for all $X \in \mathcal{B}$

iff $\chi_{(A\setminus X)} \in \mathcal{R}_o$ for all $X \in \mathcal{B}$

iff $(A \setminus X) \in \mathcal{B}$ for all $X \in \mathcal{B}$

Specified Sets and Logical Constants

A nonempty, $\mathcal{B} \subseteq \mathcal{P}$A, $\emptyset \in \mathcal{B}$, A $\in \mathcal{B}$, $\mathcal{D}$ specified sets frame.

Let $\neg : \mathcal{D}_o \longrightarrow \mathcal{D}_o$ be the negation function. $\neg \in \mathcal{D}_{o\to o}$?

$$\mathcal{R}_o = \{\chi_X | X \in \mathcal{B}\}$$
$$\mathcal{D}_{o\to o} = \{f : \mathcal{D}_o \longrightarrow \mathcal{D}_o \mid \forall p \in \mathcal{R}_o \quad (f \circ p) \in \mathcal{R}_o\}$$
$$\begin{aligned} \neg \in \mathcal{D}_{o\to o} \quad & \text{iff} \quad (\neg \circ \chi_X) \in \mathcal{R}_o \text{ for all } X \in \mathcal{B} \\ & \text{iff} \quad \chi_{(A\setminus X)} \in \mathcal{R}_o \text{ for all } X \in \mathcal{B} \\ & \text{iff} \quad (A \setminus X) \in \mathcal{B} \text{ for all } X \in \mathcal{B} \end{aligned}$$

- $\mathcal{D}$ realizes $\neg$ iff $\mathcal{B}$ is closed under complements.

Specified Sets and Logical Constants

- $\mathcal{D}$ realizes $\wedge$ iff $\mathcal{B}$ is closed under binary intersections.
- $\mathcal{D}$ realizes $\vee$ iff $\mathcal{B}$ is closed under binary unions.
- We do not have such characterizations for $=^\alpha$, Π^α and Σ^α.

Review

- Given A nonempty, $\mathcal{B} \subseteq \mathcal{P}$A, $\emptyset \in \mathcal{B}$, A $\in \mathcal{B}$.
- $\mathcal{D}_\iota = A$
- $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$
- $\mathcal{D}$ combinatory frame (evaluates terms)
- $\mathcal{D}$ may or may not realize logical constants
- Boolean Logical Constants correspond to Closure in $\mathcal{B}$

Frame in which Surjective Cantor Fails

Frame in which Surjective Cantor Fails

Let A be the real interval $(-1,1)$ and

$$\mathcal{B} := \{(a,1) \mid -1 \leq a \leq 1\} \subseteq \mathcal{P}(A).$$

Note: $\emptyset$ is $(1,1) \in \mathcal{B}$ and A is $(-1,1) \in \mathcal{B}$.

Define $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$ and $\mathcal{D}_\iota := A$.

Let $\mathcal{D}$ be Specified Sets Frame with $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$.

Frame in which Surjective Cantor Fails

Let A be the real interval $(-1,1)$ and

$$\mathcal{B} := \{(a,1) \mid -1 \leq a \leq 1\} \subseteq \mathcal{P}(A).$$

Facts:

- $\mathcal{D}_{\iota\to o} = \{\chi_{(a,1)} \mid -1 \leq a \leq 1\}$

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[1] – p.74

Frame in which Surjective Cantor Fails

Let A be the real interval $(-1,1)$ and

$$\mathcal{B} := \{(a,1) \mid -1 \leq a \leq 1\} \subseteq \mathcal{P}(A).$$

Facts:

- $\mathcal{D}_{\iota\to o} = \{\chi_{(a,1)} \mid -1 \leq a \leq 1\}$

- The negation function $\neg : \mathcal{D}_o \longrightarrow \mathcal{D}_o$ is not in $\mathcal{D}_{o\to o}$.

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[1] – p.74

Frame in which Surjective Cantor Fails

Let A be the real interval $(-1,1)$ and

$$\mathcal{B} := \{(a,1) \mid -1 \leq a \leq 1\} \subseteq \mathcal{P}(A).$$

Facts:

- $\mathcal{D}_{\iota\to o} = \{\chi_{(a,1)} \mid -1 \leq a \leq 1\}$

- The negation function $\neg : \mathcal{D}_o \longrightarrow \mathcal{D}_o$ is not in $\mathcal{D}_{o\to o}$.

- $\mathcal{D}$ realizes $\wedge$ and $\vee$.

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[1] – p.74

Frame in which Surjective Cantor Fails

Let A be the real interval $(-1, 1)$ and

$$\mathcal{B} := \{(\mathsf{a}, 1) \mid -1 \leq \mathsf{a} \leq 1\} \subseteq \mathcal{P}(\mathsf{A}).$$

Facts:

- $\mathcal{D}_{\iota\to o} = \{\chi_{(\mathsf{a},1)} \mid -1 \leq \mathsf{a} \leq 1\}$
- The negation function $\neg : \mathcal{D}_o \longrightarrow \mathcal{D}_o$ is not in $\mathcal{D}_{o\to o}$.
- $\mathcal{D}$ realizes $\wedge$ and $\vee$.
- Surjective Cantor Fails

Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$\mathcal{D}_\iota = (-1, 1)$

$\mathcal{D}_{\iota\to o} = \{\chi_{(\mathsf{a},1)} \mid -1 \leq \mathsf{a} \leq 1\}$

Define $\mathsf{G} : \mathcal{D}_\iota \longrightarrow \mathcal{D}_{\iota\to o}$ as follows:

$$\mathsf{G}(\mathsf{x}) := \begin{cases} \chi_{\emptyset} & \text{if } -1 < \mathsf{x} \leq -\frac{1}{2} \\ \chi_{(-2\mathsf{x},1)} & \text{if } -\frac{1}{2} < \mathsf{x} < \frac{1}{2} \\ \chi_{(-1,1)} & \text{if } \frac{1}{2} \leq \mathsf{x} < 1 \end{cases}$$

That is, $\mathsf{G}(\mathsf{x})(\mathsf{y}) = \mathtt{T}$ iff $-2\mathsf{x} < \mathsf{y}$ for each $\mathsf{x}, \mathsf{y} \in (-1, 1)$.

Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$\mathcal{D}_\iota = (-1, 1)$

$\mathcal{D}_{\iota\to o} = \{\chi_{(\mathsf{a},1)} \mid -1 \leq \mathsf{a} \leq 1\}$

That is, $\mathsf{G}(\mathsf{x})(\mathsf{y}) = \mathtt{T}$ iff $-2\mathsf{x} < \mathsf{y}$ for each $\mathsf{x}, \mathsf{y} \in (-1, 1)$.

Claim 1: G is surjective.

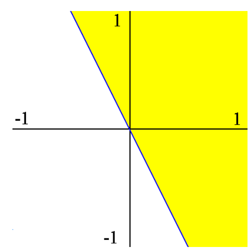
Claim 2: $\mathsf{G} \in \mathcal{D}_{\iota\to\iota\to o}$.

Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$\mathsf{G}(\mathsf{x})(\mathsf{y}) = \mathtt{T}$ iff $-2\mathsf{x} < \mathsf{y}$ for each $\mathsf{x}, \mathsf{y} \in (-1, 1)$.

G is surjective since $\mathsf{G}(-\frac{\mathsf{z}}{2})$ is $\chi_{(\mathsf{z},1)}$ for any $(\mathsf{z}, 1) \in \mathcal{B}$.

The graph of the relation $\mathsf{G}(\mathsf{x})(\mathsf{y}) = \mathtt{T}$:

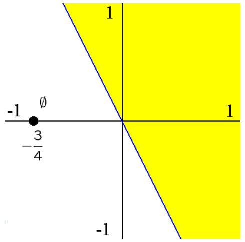


Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$G(x)(y) = T$ iff $-2x < y$ for each $x, y \in (-1, 1)$.

G is surjective since $G(-\frac{z}{2})$ is $\chi_{(z,1)}$ for any $(z, 1) \in \mathcal{B}$.

The graph of the relation $G(x)(y) = T$:

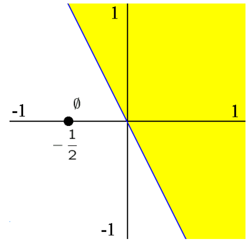


Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$G(x)(y) = T$ iff $-2x < y$ for each $x, y \in (-1, 1)$.

G is surjective since $G(-\frac{z}{2})$ is $\chi_{(z,1)}$ for any $(z, 1) \in \mathcal{B}$.

The graph of the relation $G(x)(y) = T$:

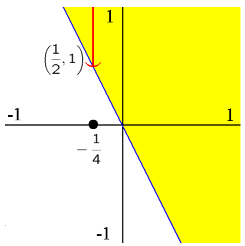


Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$G(x)(y) = T$ iff $-2x < y$ for each $x, y \in (-1, 1)$.

G is surjective since $G(-\frac{z}{2})$ is $\chi_{(z,1)}$ for any $(z, 1) \in \mathcal{B}$.

The graph of the relation $G(x)(y) = T$:

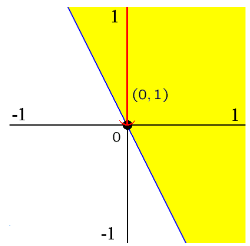


Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$G(x)(y) = T$ iff $-2x < y$ for each $x, y \in (-1, 1)$.

G is surjective since $G(-\frac{z}{2})$ is $\chi_{(z,1)}$ for any $(z, 1) \in \mathcal{B}$.

The graph of the relation $G(x)(y) = T$:

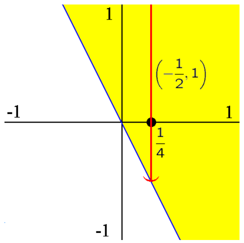


Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$G(x)(y) = T$ iff $-2x < y$ for each $x, y \in (-1, 1)$.

G is surjective since $G(-\frac{z}{2})$ is $\chi_{(z,1)}$ for any $(z, 1) \in \mathcal{B}$.

The graph of the relation $G(x)(y) = T$:

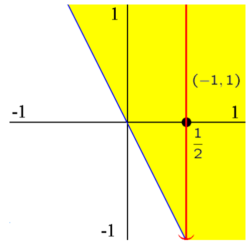


Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$G(x)(y) = T$ iff $-2x < y$ for each $x, y \in (-1, 1)$.

G is surjective since $G(-\frac{z}{2})$ is $\chi_{(z,1)}$ for any $(z, 1) \in \mathcal{B}$.

The graph of the relation $G(x)(y) = T$:

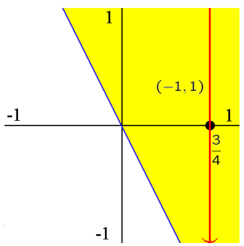


Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$G(x)(y) = T$ iff $-2x < y$ for each $x, y \in (-1, 1)$.

G is surjective since $G(-\frac{z}{2})$ is $\chi_{(z,1)}$ for any $(z, 1) \in \mathcal{B}$.

The graph of the relation $G(x)(y) = T$:



Proposed Surjection in $\mathcal{D}_{\iota\to\iota\to o}$

$G(x)(y) = T$ iff $-2x < y$ for each $x, y \in (-1, 1)$.

Is $G \in \mathcal{D}_{\iota\to\iota\to o}$?

Need to check $(G \circ p_1) \in \mathcal{R}_{\iota\to o}$ for every $p_1 \in \mathcal{R}_\iota$.

To check $(G \circ p_1) \in \mathcal{R}_{\iota\to o}$, we need to check

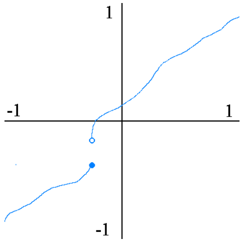
$S((G \circ p_1), p_2) \in \mathcal{R}_o$ for every $p_2 \in \mathcal{R}_\iota$.

First: Characterize $\mathcal{R}_\iota$.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \in \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

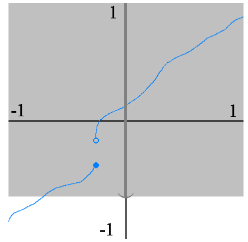
iff $p^{-1}((a,1)) \in \mathcal{B}$ for every $a \in [-1,1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \in \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

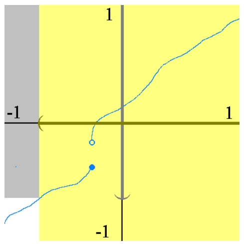
iff $p^{-1}((a,1)) \in \mathcal{B}$ for every $a \in [-1,1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \in \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

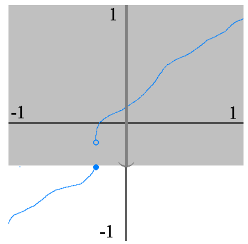
iff $p^{-1}((a,1)) \in \mathcal{B}$ for every $a \in [-1,1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \in \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

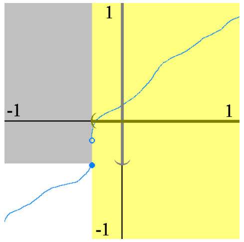
iff $p^{-1}((a,1)) \in \mathcal{B}$ for every $a \in [-1,1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \in \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

iff $p^{-1}((a, 1)) \in \mathcal{B}$ for every $a \in [-1, 1]$

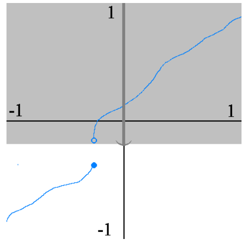
iff p is nondecreasing and left-continuous.

UNIVERSITÄT DES SAARLANDES

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \in \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

iff $p^{-1}((a, 1)) \in \mathcal{B}$ for every $a \in [-1, 1]$

iff p is nondecreasing and left-continuous.

©Benzmüller & Brown, 2006

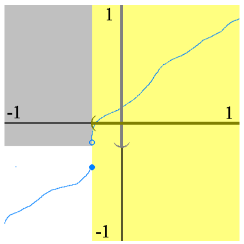
UNIVERSITÄT DES SAARLANDES

SEMHOL'06-[1] – p.78

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \in \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

iff $p^{-1}((a, 1)) \in \mathcal{B}$ for every $a \in [-1, 1]$

iff p is nondecreasing and left-continuous.

©Benzmüller & Brown, 2006

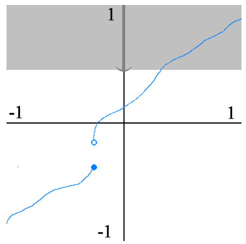
UNIVERSITÄT DES SAARLANDES

SEMHOL'06-[1] – p.78

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \in \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

iff $p^{-1}((a, 1)) \in \mathcal{B}$ for every $a \in [-1, 1]$

iff p is nondecreasing and left-continuous.

©Benzmüller & Brown, 2006

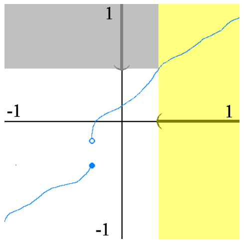
UNIVERSITÄT DES SAARLANDES

SEMHOL'06-[1] – p.78

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \in \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

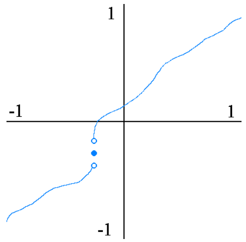
iff $p^{-1}((a,1)) \in \mathcal{B}$ for every $a \in [-1,1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \notin \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

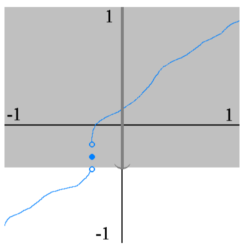
iff $p^{-1}((a,1)) \in \mathcal{B}$ for every $a \in [-1,1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \notin \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

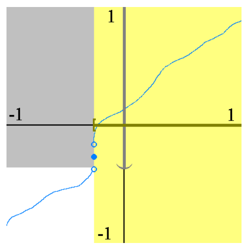
iff $p^{-1}((a,1)) \in \mathcal{B}$ for every $a \in [-1,1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \notin \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

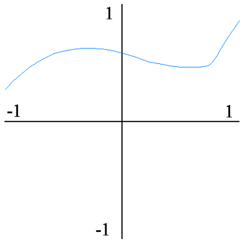
iff $p^{-1}((a,1)) \in \mathcal{B}$ for every $a \in [-1,1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \notin \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

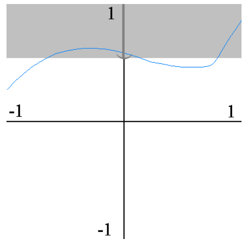
iff $p^{-1}((a, 1)) \in \mathcal{B}$ for every $a \in [-1, 1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \notin \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

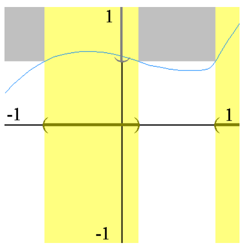
iff $p^{-1}((a, 1)) \in \mathcal{B}$ for every $a \in [-1, 1]$

iff p is nondecreasing and left-continuous.

Frame ... Surjective Cantor Fails: $\mathcal{R}_\iota$

$$\mathcal{R}_\iota = \{p : A \longrightarrow \mathcal{D}_\iota \mid \forall X \in \mathcal{B} \quad p^{-1}(X) \in \mathcal{B}\}$$

$p \notin \mathcal{R}_\iota$



Suppose $p : A \longrightarrow A$.

$p \in \mathcal{R}_\iota$

iff $p^{-1}(X) \in \mathcal{B}$ for every $X \in \mathcal{B}$

iff $p^{-1}((a, 1)) \in \mathcal{B}$ for every $a \in [-1, 1]$

iff p is nondecreasing and left-continuous.

Frame in which Surjective Cantor Fails

A is $(-1, 1)$ and $\mathcal{B}$ is $\{(a, 1) \mid -1 \leq a \leq 1\} \subseteq \mathcal{P}(A)$.

Is $G \in \mathcal{D}_{\iota \to \iota \to o}$?

Let $p_1, p_2 \in \mathcal{R}_\iota$.

$p_1, p_2 : (-1, 1) \longrightarrow (-1, 1)$ are nondecreasing and left-continuous.

Show: $S((G \circ p_1), p_2) \in^? \mathcal{R}_o$.

Need: $S((G \circ p_1), p_2)^{-1}(\mathtt{T}) \in^? \mathcal{B}$.

$$\{x \in (-1, 1) \mid S((G \circ p_1), p_2)(x) = \mathtt{T}\} \in^? \mathcal{B}$$

Frame in which Surjective Cantor Fails

Show: $\{x \in (-1,1) \mid S((G \circ p_1), p_2)(x) = T\} \in^? \mathcal{B}$

Compute:

$$S((G \circ p_1), p_2)(x) = (G \circ p_1)(x)(p_2(x)) = G(p_1(x))(p_2(x))$$

Show: $\{x \in (-1,1) \mid G(p_1(x))(p_2(x)) = T\} \in^? \mathcal{B}$

Recall: $G(x)(y) = T$ iff $-2x < y$ for each $x, y \in (-1,1)$.

Simplifies Using Definition of G:

$$\{x \in (-1,1) \mid -2p_1(x) < p_2(x)\} \in^? \mathcal{B}$$

Frame in which Surjective Cantor Fails

Show: $\{x \in (-1,1) \mid -2p_1(x) < p_2(x)\} \in^? \mathcal{B}$

Show: $\{x \in (-1,1) \mid 0 < (p_2 + 2p_1)(x)\} \in^? \mathcal{B}$

$(p_2 + 2p_1)$ is nondecreasing and left-continuous since

$p_1, p_2 \in \mathcal{R}_\iota$ are nondecreasing and left-continuous.

Thus

$$\{x \in (-1,1) \mid 0 < (p_2 + 2p_1)(x)\} = (a, 1) \in \mathcal{B}$$

for some $a \in [-1, 1]$.

Therefore, the surjection G is in $\mathcal{D}_{\iota\to\iota\to o}$ as desired.

Review

- There is a combinatory frame $\mathcal{D}$ in which Surjective Cantor Fails
- The frame does not realize $\neg$
- Negation is necessary for the diagonal argument

Injective Cantor

Injective Cantor Theorem

There is no injection from $\mathcal{P}(A)$ into A.

Injective Cantor Theorem

There is no injection from $\mathcal{P}(A)$ into A.

Proof: Assume H is an injection from $\mathcal{P}(A)$ into A.

Injective Cantor Theorem

There is no injection from $\mathcal{P}(A)$ into A.

Proof: Assume H is an injection from $\mathcal{P}(A)$ into A.

Let D be $\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}$.

Injective Cantor Theorem

There is no injection from $\mathcal{P}(A)$ into A.

Proof: Assume H is an injection from $\mathcal{P}(A)$ into A.

Let D be $\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}$.

If $H(D) \notin D$, then $H(D) \in D$ (choose $X := D$).

Injective Cantor Theorem

There is no injection from $\mathcal{P}(A)$ into A.

Proof: Assume H is an injection from $\mathcal{P}(A)$ into A.

Let D be $\{H(X) | X \in \mathcal{P}(A), H(X) \notin X\}$.

If $H(D) \notin D$, then $H(D) \in D$ (choose $X := D$).

Hence $H(D) \in D$.

Injective Cantor Theorem

There is no injection from $\mathcal{P}(A)$ into A.

Proof: Assume H is an injection from $\mathcal{P}(A)$ into A.

Let D be $\{H(X) | X \in \mathcal{P}(A), H(X) \notin X\}$.

If $H(D) \notin D$, then $H(D) \in D$ (choose $X := D$).

Hence $H(D) \in D$.

This means $H(D) = H(X)$ for some $X \in \mathcal{P}(A)$ such that $H(X) \notin X$.

Injective Cantor Theorem

There is no injection from $\mathcal{P}(A)$ into A.

Proof: Assume H is an injection from $\mathcal{P}(A)$ into A.

Let D be $\{H(X) | X \in \mathcal{P}(A), H(X) \notin X\}$.

If $H(D) \notin D$, then $H(D) \in D$ (choose $X := D$).

Hence $H(D) \in D$.

This means $H(D) = H(X)$ for some $X \in \mathcal{P}(A)$ such that $H(X) \notin X$.

H Injective $\Rightarrow D = X$.

Injective Cantor Theorem

There is no injection from $\mathcal{P}(A)$ into A.

Proof: Assume H is an injection from $\mathcal{P}(A)$ into A.

Let D be $\{H(X) | X \in \mathcal{P}(A), H(X) \notin X\}$.

If $H(D) \notin D$, then $H(D) \in D$ (choose $X := D$).

Hence $H(D) \in D$.

This means $H(D) = H(X)$ for some $X \in \mathcal{P}(A)$ such that $H(X) \notin X$.

H Injective $\Rightarrow D = X$.

Hence $H(D) \notin D$. Contradiction!

Injective Cantor in Frames

Let $\mathcal{D}$ be a frame.

Defn: We say Injective Cantor Holds in $\mathcal{D}$ if there is no injection in $\mathcal{D}_{(\iota\to o)\to\iota}$. Otherwise, we say Injective Cantor Fails.

Injective Cantor in Frames

Let $\mathcal{D}$ be a frame.

Defn: We say Injective Cantor Holds in $\mathcal{D}$ if there is no injection in $\mathcal{D}_{(\iota\to o)\to\iota}$. Otherwise, we say Injective Cantor Fails.

With more work, one can show:

Injective Cantor Fails in Specified Sets Frame $\mathcal{D}$ given by $(-1,1)$ and $\{(a,1)|a \in [-1,1]\}$.

There is an injection $H \in \mathcal{D}_{(\iota\to o)\to\iota}$.

In fact, $G \circ H$ is the identity function.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing , , , and . Injective Cantor Holds in $\mathcal{D}$.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing , , , and . Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $H \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing , , , and .
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Need “internal” version of diagonal set

$$''\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}''$$

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing , , , and .
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Need “internal” version of diagonal set

$$''\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}''$$

Characteristic Function Of Set:

$$\{H(X)|X \in \mathcal{D}_{\iota\to o} X(H(X)) = \mathtt{F}\}$$

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing , , , and .
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Need “internal” version of diagonal set

$$''\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}''$$

Characteristic Function Of Set:

$$\{Y|\exists X \in \mathcal{D}_{\iota\to o} Y = H(X) \wedge X(H(X)) = \mathtt{F}\}$$

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing , , , and .
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Need “internal” version of diagonal set

$$''\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}''$$

Characteristic Function Of Set:

$$\{Y|\exists X \in \mathcal{D}_{\iota\to o} Y = H(X) \wedge X(H(X)) = \mathtt{F}\}$$

Evaluation of Term:

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y{=}^{\iota} H(X) \wedge \neg[X[HX]]]]$$

where $\varphi(H) = H$.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing , , $\Sigma^{\iota\to o}$, and .
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Need “internal” version of diagonal set

$$''\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}''$$

Characteristic Function Of Set:

$$\{Y|\exists X \in \mathcal{D}_{\iota\to o} Y = H(X) \wedge X(H(X)) = \mathtt{F}\}$$

Evaluation of Term:

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y{=}^{\iota}H(X){\wedge}\neg[X[HX]]]]$$

where $\varphi(H) = H$.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing , , $\Sigma^{\iota\to o}$, and $=^{\iota}$.
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Need “internal” version of diagonal set

$$''\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}''$$

Characteristic Function Of Set:

$$\{Y|\exists X \in \mathcal{D}_{\iota\to o} Y = H(X) \wedge X(H(X)) = \mathtt{F}\}$$

Evaluation of Term:

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y{=}^{\iota}H(X){\wedge}\neg[X[HX]]]]$$

where $\varphi(H) = H$.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing , $\wedge$, $\Sigma^{\iota\to o}$, and $=^{\iota}$.
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Need “internal” version of diagonal set

$$''\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}''$$

Characteristic Function Of Set:

$$\{Y|\exists X \in \mathcal{D}_{\iota\to o} Y = H(X) \wedge X(H(X)) = \mathtt{F}\}$$

Evaluation of Term:

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y{=}^{\iota}H(X){\wedge}\neg[X[HX]]]]$$

where $\varphi(H) = H$.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing $\neg$, $\wedge$, $\Sigma^{\iota\to o}$, and $=^{\iota}$.
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Need “internal” version of diagonal set

$$''\{H(X)|X \in \mathcal{P}(A), H(X) \notin X\}''$$

Characteristic Function Of Set:

$$\{Y|\exists X \in \mathcal{D}_{\iota\to o} Y = H(X) \wedge X(H(X)) = \mathtt{F}\}$$

Evaluation of Term:

$$[\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y{=}^{\iota}H(X){\wedge}\neg[X[HX]]]]$$

where $\varphi(H) = H$.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing $\neg$, $\wedge$, $\Sigma^{\iota\to o}$, and $=^{\iota}$.
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Let Σ be $\{\neg, \wedge, \Sigma^{\iota\to o}, =^{\iota}\}$.

Let $\mathcal{I} : (\Sigma \cup \mathcal{P}) \longrightarrow \mathcal{D}$ such that $\mathfrak{L}_{\mathsf{c}}(\mathcal{I}(\mathsf{c}))$ for $\mathsf{c} \in \Sigma$.

$\mathcal{D}$ combinatory $\Rightarrow \exists\mathcal{E}$ evaluating terms extending $\mathcal{I}$

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing $\neg$, $\wedge$, $\Sigma^{\iota\to o}$, and $=^{\iota}$.
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Let D be

$$\mathcal{E}_{\varphi}([\lambda \mathsf{Y}_{\iota\to o}.\exists \mathsf{X}_{\iota\to o}[\mathsf{Y}=^{\iota}\mathsf{H}(\mathsf{X})\wedge\neg[\mathsf{X}\,[\mathsf{H}\,\mathsf{X}]]]])$$

where $\varphi(\mathsf{H}) = \mathsf{H}$.

If $\mathsf{D}(\mathsf{H}(\mathsf{D})) = \mathtt{F}$, then $\mathsf{D}(\mathsf{H}(\mathsf{D})) = \mathtt{T}$.
Hence $\mathsf{D}(\mathsf{H}(\mathsf{D}))=\mathtt{T}$.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing $\neg$, $\wedge$, $\Sigma^{\iota\to o}$, and $=^{\iota}$.
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Let D be

$$\mathcal{E}_{\varphi}([\lambda \mathsf{Y}_{\iota\to o}.\exists \mathsf{X}_{\iota\to o}[\mathsf{Y}=^{\iota}\mathsf{H}(\mathsf{X})\wedge\neg[\mathsf{X}\,[\mathsf{H}\,\mathsf{X}]]]])$$

where $\varphi(\mathsf{H}) = \mathsf{H}$.

If $\mathsf{D}(\mathsf{H}(\mathsf{D})) = \mathtt{F}$, then $\mathsf{D}(\mathsf{H}(\mathsf{D})) = \mathtt{T}$.
Hence $\mathsf{D}(\mathsf{H}(\mathsf{D}))=\mathtt{T}$.
Hence $\mathsf{H}(\mathsf{D})$ equals $\mathsf{H}(\mathsf{X})$ where $\mathsf{X}(\mathsf{H}(\mathsf{X}))=\mathtt{F}$ for some X.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing $\neg$, $\wedge$, $\Sigma^{\iota\to o}$, and $=^{\iota}$.
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $\mathsf{H} \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Let D be

$$\mathcal{E}_{\varphi}([\lambda \mathsf{Y}_{\iota\to o}.\exists \mathsf{X}_{\iota\to o}[\mathsf{Y}=^{\iota}\mathsf{H}(\mathsf{X})\wedge\neg[\mathsf{X}\,[\mathsf{H}\,\mathsf{X}]]]])$$

where $\varphi(\mathsf{H}) = \mathsf{H}$.

If $\mathsf{D}(\mathsf{H}(\mathsf{D})) = \mathtt{F}$, then $\mathsf{D}(\mathsf{H}(\mathsf{D})) = \mathtt{T}$.
Hence $\mathsf{D}(\mathsf{H}(\mathsf{D}))=\mathtt{T}$.
Hence $\mathsf{H}(\mathsf{D})$ equals $\mathsf{H}(\mathsf{X})$ where $\mathsf{X}(\mathsf{H}(\mathsf{X}))=\mathtt{F}$ for some X.

H injective implies D equals X.

Injective Cantor in Frames

Let $\mathcal{D}$ be a combinatory frame realizing $\neg$, $\wedge$, $\Sigma^{\iota\to o}$, and $=^{\iota}$.
Injective Cantor Holds in $\mathcal{D}$.

Proof: Assume $H \in \mathcal{D}_{(\iota\to o)\longrightarrow\iota}$ is injective.

Let D be

$$\mathcal{E}_{\varphi}([\lambda Y_{\iota\to o}.\exists X_{\iota\to o}[Y{=}^{\iota}H(X)\wedge\neg[X\,[H\,X]]]])$$

where $\varphi(H) = H$.

If $D(H(D)) = F$, then $D(H(D)) = T$.
Hence $D(H(D)){=}T$.
Hence $H(D)$ equals $H(X)$ where $X(H(X)){=}F$ for some X.

H injective implies D equals X. Contradiction.

Injective Cantor Theorem

Claim: There is a combinatory frame $\mathcal{D}$ such that:

- Injective Cantor Fails
- Surjective Cantor Holds
- $\mathcal{D}$ realizes $\neg$ and $\wedge$. ($\Rightarrow$ Surjective Cantor Holds)
- $\mathcal{D}$ realizes neither $=^{\iota}$ nor $\Sigma^{\iota\to o}$.

Frame in which Injective Cantor Fails

Let A be the natural numbers $\mathbf{N}$.

Let $\mathcal{B}$ be the set of finite $X \subseteq \mathbf{N}$ and cofinite $Y \subseteq \mathbf{N}$.

Note: $\emptyset \in \mathcal{B}$ (finite), $\mathbf{N} \in \mathcal{B}$ (cofinite), and $\mathcal{B}$ is closed

under complements, binary unions and binary intersections.

Specified Sets Frame: $\mathcal{D}_o := \{T, F\}$, $\mathcal{D}_\iota := \mathbf{N}$, and $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$.

- $\mathcal{D}$ realizes $\neg$, $\wedge$, and $\vee$

Frame in which Injective Cantor Fails

Let A be the natural numbers $\mathbf{N}$.

Let $\mathcal{B}$ be the set of finite $X \subseteq \mathbf{N}$ and cofinite $Y \subseteq \mathbf{N}$.

Specified Sets Frame: $\mathcal{D}_o := \{T, F\}$, $\mathcal{D}_\iota := \mathbf{N}$, and $\mathcal{D}_{\iota\to o} \cong \mathcal{B}$.

- $\mathcal{D}$ realizes $\neg$, $\wedge$, and $\vee$
- Both $\mathcal{D}_\iota$ and $\mathcal{D}_{\iota\to o}$ are countable.
- There is a bijection.

Frame in which Injective Cantor Fails

Let A be the natural numbers $\mathbf{N}$.

Let $\mathcal{B}$ be the set of finite $X \subseteq \mathbf{N}$ and cofinite $Y \subseteq \mathbf{N}$.

Specified Sets Frame: $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota := \mathbf{N}$, and $\mathcal{D}_{\iota \to o} \cong \mathcal{B}$.

- $\mathcal{D}$ realizes $\neg$, $\wedge$, and $\vee$
- Both $\mathcal{D}_\iota$ and $\mathcal{D}_{\iota \to o}$ are countable.
- There is a bijection.
- Already know Surjective Cantor Holds $\Rightarrow$ There is no surjection.

Frame in which Injective Cantor Fails

Let A be the natural numbers $\mathbf{N}$.

Let $\mathcal{B}$ be the set of finite $X \subseteq \mathbf{N}$ and cofinite $Y \subseteq \mathbf{N}$.

Specified Sets Frame: $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota := \mathbf{N}$, and $\mathcal{D}_{\iota \to o} \cong \mathcal{B}$.

- $\mathcal{D}$ realizes $\neg$, $\wedge$, and $\vee$
- Both $\mathcal{D}_\iota$ and $\mathcal{D}_{\iota \to o}$ are countable.
- There is a bijection.
- Already know Surjective Cantor Holds $\Rightarrow$ There is no surjection.
- "Skolem's Paradox"

Frame in which Injective Cantor Fails

Let A be the natural numbers $\mathbf{N}$.

Let $\mathcal{B}$ be the set of finite $X \subseteq \mathbf{N}$ and cofinite $Y \subseteq \mathbf{N}$.

Specified Sets Frame: $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota := \mathbf{N}$, and $\mathcal{D}_{\iota \to o} \cong \mathcal{B}$.

- Need Surjection/Injection to be *in* the frame.
- No Surjection is in the frame.
- Injection?

Frame in which Injective Cantor Fails

Let A be the natural numbers $\mathbf{N}$.

Let $\mathcal{B}$ be the set of finite $X \subseteq \mathbf{N}$ and cofinite $Y \subseteq \mathbf{N}$.

Specified Sets Frame: $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$, $\mathcal{D}_\iota := \mathbf{N}$, and $\mathcal{D}_{\iota \to o} \cong \mathcal{B}$.

- Need Surjection/Injection to be *in* the frame.
- No Surjection is in the frame.
- Injection? All the injections will be in the frame...

Frame ... Injective Cantor Fails – $\mathcal{R}_o$

$\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$

$\mathcal{D}_\iota = \mathbf{N}$

$\mathcal{R}_o = \{\chi_X : \mathbf{N} \longrightarrow \mathcal{D}_o \mid X \text{ finite or cofinite}\}$

Frame ... Injective Cantor Fails – $\mathcal{R}_o$

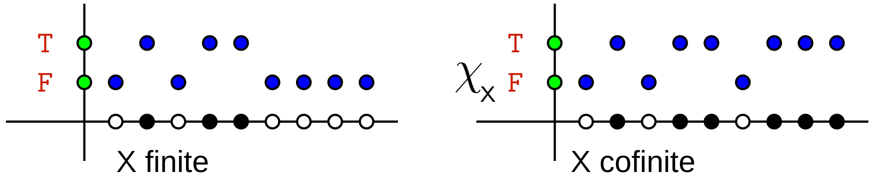
$\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$

$\mathcal{D}_\iota = \mathbf{N}$

$\mathcal{R}_o = \{\chi_X : \mathbf{N} \longrightarrow \mathcal{D}_o \mid X \text{ finite or cofinite}\}$

Two Cases:

1. X is finite and χ_X is eventually $\mathtt{F}$.
2. X is cofinite and χ_X is eventually $\mathtt{T}$.



Frame ... Injective Cantor Fails – $\mathcal{R}_o$

$\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$

$\mathcal{D}_\iota = \mathbf{N}$

$\mathcal{R}_o = \{\chi_X : \mathbf{N} \longrightarrow \mathcal{D}_o \mid X \text{ finite or cofinite}\}$

Two Cases:

1. X is finite and χ_X is eventually $\mathtt{F}$.
2. X is cofinite and χ_X is eventually $\mathtt{T}$.

Lemma: $p \in \mathcal{R}_o$ iff $p : \mathbf{N} \longrightarrow \mathcal{D}_o$ is eventually constant.

Frame ... Injective Cantor Fails – $\mathcal{R}_o$

$\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$

$\mathcal{D}_\iota = \mathbf{N}$

$\mathcal{R}_o = \{\chi_X : \mathbf{N} \longrightarrow \mathcal{D}_o \mid X \text{ finite or cofinite}\}$

Two Cases:

1. X is finite and χ_X is eventually $\mathtt{F}$.
2. X is cofinite and χ_X is eventually $\mathtt{T}$.

Lemma: $p \in \mathcal{R}_o$ iff $p : \mathbf{N} \longrightarrow \mathcal{D}_o$ is eventually constant.

Defn: A function $f : \mathbf{N} \longrightarrow C$ is eventually constant if there is some $N \in \mathbf{N}$ and $c \in C$ such that $f(n) = c$ for every $n \geq N$.

Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Suppose $p \in \mathcal{R}_\iota$.

p is a sequence of natural numbers.

Inverse images of finite/cofinite sets are either finite or cofinite.

Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

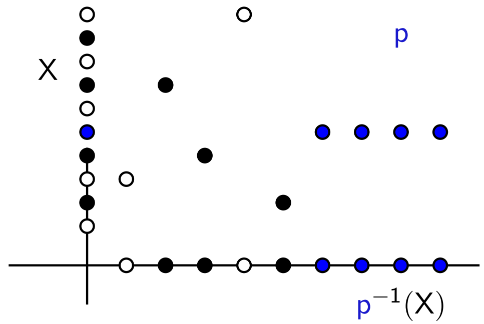
$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Suppose $p \in \mathcal{R}_\iota$.

p is a sequence of natural numbers.

Inverse images of finite/cofinite sets are either finite or cofinite.

- Eventually Constant p satisfy this.



Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

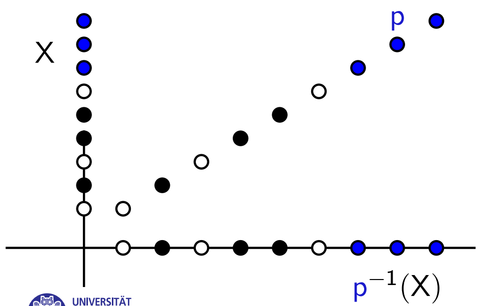
$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Suppose $p \in \mathcal{R}_\iota$.

p is a sequence of natural numbers.

Inverse images of finite/cofinite sets are either finite or cofinite.

- Eventually Constant p satisfy this.
- So does identity function – not eventually constant.



Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Suppose $p \in \mathcal{R}_\iota$.

p is a sequence of natural numbers.

Inverse images of finite/cofinite sets are either finite or cofinite.

- Eventually Constant p satisfy this.
- So does identity function – not eventually constant.

Defn: We say p is uniformly unbounded if for every $j \in \mathbf{N}$ there is some $I \in \mathbf{N}$ such that $p(i) > j$ for every $i > I$.

Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Lemma: Let $p : \mathbf{N} \longrightarrow \mathbf{N}$.
$p \in \mathcal{R}_\iota$ iff p is either eventually constant or uniformly unbounded.

Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Lemma: Let $p : \mathbf{N} \longrightarrow \mathbf{N}$.
$p \in \mathcal{R}_\iota$ iff p is either eventually constant or uniformly unbounded.

Proof: "$\Rightarrow$" Assume $p \in \mathcal{R}_\iota$ and p is not eventually constant.
Check: p is uniformly unbounded.
Let $j \in \mathbf{N}$ be given. $p^{-1}(\{0, ..., j\})$ is finite or cofinite.

Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Lemma: Let $p : \mathbf{N} \longrightarrow \mathbf{N}$.
$p \in \mathcal{R}_\iota$ iff p is either eventually constant or uniformly unbounded.

Proof: "$\Rightarrow$" Assume $p \in \mathcal{R}_\iota$ and p is not eventually constant.
Check: p is uniformly unbounded.
Let $j \in \mathbf{N}$ be given. $p^{-1}(\{0, ..., j\})$ is finite or cofinite.

• Assume $p^{-1}(\{0, ..., j\})$ is cofinite
$\Rightarrow \exists L \in \{0, ..., j\} \ni p^{-1}(\{L\})$ is cofinite
$\Rightarrow$ p is eventually L, contradicting p not eventually constant.

Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Lemma: Let $p : \mathbf{N} \longrightarrow \mathbf{N}$.
$p \in \mathcal{R}_\iota$ iff p is either eventually constant or uniformly unbounded.

Proof: "$\Rightarrow$" Assume $p \in \mathcal{R}_\iota$ and p is not eventually constant.
Check: p is uniformly unbounded.
Let $j \in \mathbf{N}$ be given. $p^{-1}(\{0, ..., j\})$ is finite or cofinite.

• $p^{-1}(\{0, ..., j\})$ is finite
Hence $\subseteq \{0, ..., l\}$ for some $l \in \mathbf{N}$.

Thus $p(i) > j$ for all $i > l$.

Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Lemma: Let $p : \mathbf{N} \longrightarrow \mathbf{N}$.
$p \in \mathcal{R}_\iota$ iff p is either eventually constant or uniformly unbounded.

Proof: "$\Leftarrow$" p eventually constant
$\Rightarrow p^{-1}(X)$ is finite or cofinite (based on whether the eventual value is in X)

$\Rightarrow p \in \mathcal{R}_\iota$

Frame ... Injective Cantor Fails – $\mathcal{R}_\iota$

$$\mathcal{R}_\iota := \{p : \mathbf{N} \longrightarrow \mathbf{N} | p^{-1}(X) \in \mathcal{B} \forall X \in \mathcal{B}\}$$

Lemma: Let $p : \mathbf{N} \longrightarrow \mathbf{N}$.
$p \in \mathcal{R}_\iota$ iff p is either eventually constant or uniformly unbounded.

Proof: "$\Leftarrow$" p uniformly unbounded
$\Rightarrow p^{-1}(X)$ is finite if X if finite
and $p^{-1}(X)$ is cofinite if X is cofinite

$\Rightarrow p \in \mathcal{R}_\iota$

Eventually Equal Functions in $\mathcal{R}$

Defn: Functions $p : \mathbf{N} \longrightarrow C$ and $q : \mathbf{N} \longrightarrow C$ are eventually equal if there is some $N \in \mathbf{N}$ such that $p(n) = q(n)$ for every $n \geq N$.

Eventually Equal Functions in $\mathcal{R}$

Defn: Functions $p : \mathbf{N} \longrightarrow C$ and $q : \mathbf{N} \longrightarrow C$ are eventually equal if there is some $N \in \mathbf{N}$ such that $p(n) = q(n)$ for every $n \geq N$.

Lemma: $p, q : \mathbf{N} \longrightarrow \mathcal{D}_\alpha$
If $p \in \mathcal{R}_\alpha$ and p and q are eventually equal, then $q \in \mathcal{R}_\alpha$.

Eventually Equal Functions in $\mathcal{R}$

Defn: Functions $p : \mathbf{N} \longrightarrow C$ and $q : \mathbf{N} \longrightarrow C$ are eventually equal if there is some $N \in \mathbf{N}$ such that $p(n) = q(n)$ for every $n \geq N$.

Lemma: $p, q : \mathbf{N} \longrightarrow \mathcal{D}_\alpha$
If $p \in \mathcal{R}_\alpha$ and p and q are eventually equal, then $q \in \mathcal{R}_\alpha$.

Proof: Induction on type α
$\alpha = o$: $p \in \mathcal{R}_o$
$\Rightarrow$ p eventually constant
$\Rightarrow$ q eventually constant
$\Rightarrow$ $q \in \mathcal{R}_o$.

Eventually Equal Functions in $\mathcal{R}$

Defn: Functions $p : \mathbf{N} \longrightarrow C$ and $q : \mathbf{N} \longrightarrow C$ are eventually equal if there is some $N \in \mathbf{N}$ such that $p(n) = q(n)$ for every $n \geq N$.

Lemma: $p, q : \mathbf{N} \longrightarrow \mathcal{D}_\alpha$
If $p \in \mathcal{R}_\alpha$ and p and q are eventually equal, then $q \in \mathcal{R}_\alpha$.

Proof: Induction on type α
$\alpha = \iota$: $p \in \mathcal{R}_\iota$
$\Rightarrow$ p eventually constant or uniformly unbounded
$\Rightarrow$ q eventually constant or uniformly unbounded
$\Rightarrow$ $q \in \mathcal{R}_\iota$.

Eventually Equal Functions in $\mathcal{R}$

Defn: Functions $p : \mathbf{N} \longrightarrow C$ and $q : \mathbf{N} \longrightarrow C$ are eventually equal if there is some $N \in \mathbf{N}$ such that $p(n) = q(n)$ for every $n \geq N$.

Lemma: $p, q : \mathbf{N} \longrightarrow \mathcal{D}_\alpha$
If $p \in \mathcal{R}_\alpha$ and p and q are eventually equal, then $q \in \mathcal{R}_\alpha$.

Proof: Induction on type α
$\alpha = (\beta \to \gamma)$: Let $r \in \mathcal{R}_\beta$. Check $S(q, r) \in \mathcal{R}_\gamma$.
Know: $S(p, r) \in \mathcal{R}_\gamma$.

Eventually Equal Functions in $\mathcal{R}$

Defn: Functions $p : \mathbf{N} \longrightarrow C$ and $q : \mathbf{N} \longrightarrow C$ are eventually equal if there is some $N \in \mathbf{N}$ such that $p(n) = q(n)$ for every $n \geq N$.

Lemma: $p, q : \mathbf{N} \longrightarrow \mathcal{D}_\alpha$
If $p \in \mathcal{R}_\alpha$ and p and q are eventually equal, then $q \in \mathcal{R}_\alpha$.

Proof: Induction on type α
$\alpha = (\beta \to \gamma)$: Let $r \in \mathcal{R}_\beta$. Check $S(q, r) \in \mathcal{R}_\gamma$.
Know: $S(p, r) \in \mathcal{R}_\gamma$.
For big enough $n \in \mathbf{N}$, $p(n) = q(n)$ and so

$$S(p, r)(n) = p(n)(r(n)) = q(n)(r(n)) = S(q, r)(n)$$

Eventually Equal Functions in $\mathcal{R}$

Defn: Functions $p : \mathbf{N} \longrightarrow C$ and $q : \mathbf{N} \longrightarrow C$ are eventually equal if there is some $N \in \mathbf{N}$ such that $p(n) = q(n)$ for every $n \geq N$.

Lemma: $p, q : \mathbf{N} \longrightarrow \mathcal{D}_\alpha$
If $p \in \mathcal{R}_\alpha$ and p and q are eventually equal, then $q \in \mathcal{R}_\alpha$.

Proof: Induction on type α
$\alpha = (\beta \rightarrow \gamma)$: Let $r \in \mathcal{R}_\beta$. Check $S(q, r) \in \mathcal{R}_\gamma$.
Know: $S(p, r) \in \mathcal{R}_\gamma$.
$S(p, r)$ and $S(q, r)$ are eventually equal.

Inductive hypothesis (at γ) $\Rightarrow S(q, r) \in \mathcal{R}_\gamma$ QED

Eventually Constant Functions in $\mathcal{R}$

Lemma: For any type α, if $p : \mathbf{N} \longrightarrow \mathcal{D}_\alpha$ is eventually constant, then $p \in \mathcal{R}_\alpha$.
Proof: p is eventually equal to a constant function.
Constant functions are in $\mathcal{R}_\alpha$. QED

Every eventually constant function is in $\mathcal{R}_\alpha$.

For some α, $\mathcal{R}_\alpha$ contains *exactly* the eventually constant functions.

For such α, we have...

Full Domains

Fact: If every $p \in \mathcal{R}_\alpha$ is eventually constant, then $\mathcal{D}_{\alpha\rightarrow\beta} = (\mathcal{D}_\beta)^{\mathcal{D}_\alpha}$ for every type β.
Proof: Know $\mathcal{D}_{\alpha\rightarrow\beta} \subseteq (\mathcal{D}_\beta)^{\mathcal{D}_\alpha}$. Show: $(\mathcal{D}_\beta)^{\mathcal{D}_\alpha} \subseteq \mathcal{D}_{\alpha\rightarrow\beta}$.
Let $f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$ be given.
Let $p \in \mathcal{R}_\alpha$ be given. By assumption, p is eventually constant.
Hence $(f \circ p)$ is eventually constant.
Hence $(f \circ p) \in \mathcal{R}_\beta$.

Hence $f \in \mathcal{D}_{\alpha\rightarrow\beta}$.

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\rightarrow o}$

Lemma: Every $p \in \mathcal{R}_{\iota\rightarrow o}$ is eventually constant.

- After showing this, we know $\mathcal{D}_{(\iota\rightarrow o)\rightarrow\beta}$ contains all functions from $\mathcal{D}_{\iota\rightarrow o}$ to $\mathcal{D}_\beta$.

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
We first prove:

$$(1)\exists M, N \in \mathbf{N}. p(m)(n) = p(M)(N) \forall m > M, n > N$$

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
We first prove:

$$(1)\exists M, N \in \mathbf{N}. p(m)(n) = p(M)(N) \forall m > M, n > N$$

Otherwise, have $0 = m_0 < m_1 < \cdots$ and $0 = n_0 < n_1 < \cdots$ such that $p(m_{i+1})(n_{i+1}) \neq p(m_i)(n_i)$.

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
We first prove:

$$(1)\exists M, N \in \mathbf{N}. p(m)(n) = p(M)(N) \forall m > M, n > N$$

Otherwise, have $0 = m_0 < m_1 < \cdots$ and $0 = n_0 < n_1 < \cdots$ such that $p(m_{i+1})(n_{i+1}) \neq p(m_i)(n_i)$.
Define $q : \mathbf{N} \longrightarrow \mathbf{N}$ by $q(m) := n_l$ where l is least i with $m_i \leq m < m_{i+1}$.

q is uniformly unbounded

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
We first prove:

$$(1)\exists M, N \in \mathbf{N}. p(m)(n) = p(M)(N) \forall m > M, n > N$$

Otherwise, have $0 = m_0 < m_1 < \cdots$ and $0 = n_0 < n_1 < \cdots$ such that $p(m_{i+1})(n_{i+1}) \neq p(m_i)(n_i)$.
Define $q : \mathbf{N} \longrightarrow \mathbf{N}$ by $q(m) := n_l$ where l is least i with $m_i \leq m < m_{i+1}$.
q is uniformly unbounded $\Rightarrow q \in \mathcal{R}_\iota$.

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
We first prove:

$$(1)\exists M, N \in \mathbf{N}.p(m)(n) = p(M)(N)\forall m > M, n > N$$

Otherwise, have $0 = m_0 < m_1 < \cdots$ and $0 = n_0 < n_1 < \cdots$ such that $p(m_{i+1})(n_{i+1}) \neq p(m_i)(n_i)$.
Define $q : \mathbf{N} \longrightarrow \mathbf{N}$ by $q(m) := n_l$ where l is least i with $m_i \leq m < m_{i+1}$.
q is uniformly unbounded $\Rightarrow q \in \mathcal{R}_\iota$.

$S(p, q) \in \mathcal{R}_o$

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
We first prove:

$$(1)\exists M, N \in \mathbf{N}.p(m)(n) = p(M)(N)\forall m > M, n > N$$

Otherwise, have $0 = m_0 < m_1 < \cdots$ and $0 = n_0 < n_1 < \cdots$ such that $p(m_{i+1})(n_{i+1}) \neq p(m_i)(n_i)$.
Define $q : \mathbf{N} \longrightarrow \mathbf{N}$ by $q(m) := n_l$ where l is least i with $m_i \leq m < m_{i+1}$.
q is uniformly unbounded $\Rightarrow q \in \mathcal{R}_\iota$.
$S(p, q) \in \mathcal{R}_o$ $\Rightarrow S(p, q)$ eventually constant

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
We first prove:

$$(1)\exists M, N \in \mathbf{N}.p(m)(n) = p(M)(N)\forall m > M, n > N$$

Otherwise, have $0 = m_0 < m_1 < \cdots$ and $0 = n_0 < n_1 < \cdots$ such that $p(m_{i+1})(n_{i+1}) \neq p(m_i)(n_i)$.
Define $q : \mathbf{N} \longrightarrow \mathbf{N}$ by $q(m) := n_l$ where l is least i with $m_i \leq m < m_{i+1}$.
q is uniformly unbounded $\Rightarrow q \in \mathcal{R}_\iota$.
$S(p, q) \in \mathcal{R}_o$ $\Rightarrow S(p, q)$ eventually constant

However, $S(p, q)(m_i) = p(m_i)(n_i)$ alternates by construction.

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
So we have $M, N \in \mathbf{N}$ such that

$$p(m)(n) = p(M)(N)\forall m > M, n > N$$

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
So we have $M, N \in \mathbf{N}$ such that

$$p(m)(n) = p(M)(N) \forall m > M, n > N$$

Now: For each $n \in \{0, \ldots, N\}$ there is some M_n and a_n such that $p(m)(n) = a_n$ for $m > M_n$.

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
So we have $M, N \in \mathbf{N}$ such that

$$p(m)(n) = p(M)(N) \forall m > M, n > N$$

Now: For each $n \in \{0, \ldots, N\}$ there is some M_n and a_n such that $p(m)(n) = a_n$ for $m > M_n$.
$S(p, K_n) \in \mathcal{R}_o$ must be eventually constant,

and $p(m)(n) = S(p, K_n)(m)$.

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
So we have $M, N \in \mathbf{N}$ such that

$$p(m)(n) = p(M)(N) \forall m > M, n > N$$

Now: For each $n \in \{0, \ldots, N\}$ there is some M_n and a_n such that $p(m)(n) = a_n$ for $m > M_n$.
Let M' be $\max(M, M_0, \ldots, M_N)$.
Claim: For all $m > M'$, $p(m) = p(M' + 1)$.

Frame ... Injective Cantor Fails – $\mathcal{R}_{\iota\to o}$

Lemma: Every $p \in \mathcal{R}_{\iota\to o}$ is eventually constant.

Proof:
So we have $M, N \in \mathbf{N}$ such that

$$p(m)(n) = p(M)(N) \forall m > M, n > N$$

Now: For each $n \in \{0, \ldots, N\}$ there is some M_n and a_n such that $p(m)(n) = a_n$ for $m > M_n$.
Let M' be $\max(M, M_0, \ldots, M_N)$.
Claim: For all $m > M'$, $p(m) = p(M' + 1)$.
If $n \leq N$, $p(m)(n) = p(M' + 1)(n)$ since $M' \geq M_n$.
If $n > N$, $p(m)(n) = p(M' + 1)(n)$ since $M' \geq M$.

Hence p is eventually constant. QED

Frame in which Injective Cantor Fails

Conclusions:

- $f \in \mathcal{D}_{(\iota\to o)\to\iota}$ for every $f : \mathcal{D}_{\iota\to o} \longrightarrow \mathcal{D}_\iota$.
- Every injection from $\mathcal{D}_{\iota\to o}$ to $\mathcal{D}_\iota$ (both are countable) is in $\mathcal{D}_{(\iota\to o)\to\iota}$.
- Injective Cantor Fails in $\mathcal{D}$.
- $f \in \mathcal{D}_{(\iota\to o)\to o}$ for every $f : \mathcal{D}_{\iota\to o} \longrightarrow \mathcal{D}_o$.
- Π^ι and Σ^ι have interpretations in $\mathcal{D}_{(\iota\to o)\to o}$.

Contravariant Effects

- If $\mathcal{R}_\alpha \subseteq (\mathcal{D}_\alpha)^A$ is **small**, then the sets

$$\mathcal{D}_{\alpha\to o} := \{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_o \mid \forall p \in \boxed{\mathcal{R}_\alpha} \quad (f \circ p) \in \mathcal{R}_o\}$$

and

$$\mathcal{R}_{\alpha\to o} := \{q : A \longrightarrow \mathcal{D}_{\alpha\to o} \mid \forall p \in \boxed{\mathcal{R}_\alpha} \quad S(q,p) \in \mathcal{R}_o\}$$

are **big** (fewer restrictions).

- If $\mathcal{R}_\alpha \subseteq (\mathcal{D}_\alpha)^A$ is **big**, then the sets

$$\mathcal{D}_{\alpha\to o} := \{f : \mathcal{D}_\alpha \longrightarrow \mathcal{D}_o \mid \forall p \in \boxed{\mathcal{R}_\alpha} \quad (f \circ p) \in \mathcal{R}_o\}$$

and

$$\mathcal{R}_{\alpha\to o} := \{q : A \longrightarrow \mathcal{D}_{\alpha\to o} \mid \forall p \in \boxed{\mathcal{R}_\alpha} \quad S(q,p) \in \mathcal{R}_o\}$$

are **small** (more restrictions).

Contravariant Effects

- $\mathcal{R}_\iota$ is **big**.
- $\mathcal{D}_{\iota\to o}$ and $\mathcal{R}_{\iota\to o}$ are **small**.
- $\mathcal{D}_{(\iota\to o)\to o}$ and $\mathcal{R}_{(\iota\to o)\to o}$ are **big**.
- $\mathcal{D}_{((\iota\to o)\to o)\to o}$ and $\mathcal{R}_{((\iota\to o)\to o)\to o}$ are **small**.
- $\mathcal{D}_{(((\iota\to o)\to o)\to o)\to o}$ and $\mathcal{R}_{(((\iota\to o)\to o)\to o)\to o}$ are **big**.
- etc. . .

Frame in which Injective Cantor Fails

- $\mathcal{D}_\iota$ and $\mathcal{D}_{\iota\to o}$ are countable.
- $\mathcal{D}_{(\iota\to o)\to o} = (\mathcal{D}_o)^{\mathcal{D}_{\iota\to o}}$ is uncountable (continuum).
- $\mathcal{D}_{((\iota\to o)\to o)\to o}$ is again *countable*.
- $\mathcal{D}_{(((\iota\to o)\to o)\to o)\to o} = (\mathcal{D}_o)^{\mathcal{D}_{((\iota\to o)\to o)\to o}}$ is uncountable (continuum).
- Alternates up the heirarchy.

Frame in which Injective Cantor Fails

	cardinality	standard	Π^α	Σ^α	$=^\alpha$
$\vdots$					
$\mathcal{D}_{((((\iota\to o)\to o)\to o)\to o)\to o}$	$\aleph_0$	no	no	no	yes
$\mathcal{D}_{(((\iota\to o)\to o)\to o)\to o}$	$2^{\aleph_0}$	yes	yes	yes	no
$\mathcal{D}_{((\iota\to o)\to o)\to o}$	$\aleph_0$	no	no	no	yes
$\mathcal{D}_{(\iota\to o)\to o}$	$2^{\aleph_0}$	yes	yes	yes	no
$\mathcal{D}_{\iota\to o}$	$\aleph_0$	no	no	no	yes
$\mathcal{D}_{\iota}$	$\aleph_0$	–	yes	yes	no

Review

- Realizing $\neg$, $\wedge$, $\Sigma^{\iota\to o}$ and $=^\iota$ is enough for Injective Cantor to Hold
- Realizing $\neg$ and $\wedge$ are not enough

Generalizing the Semantics

More Syntax

- α-conversion: We consider terms "identical" if they are the same up renaming of bound variables.

Example: $[\lambda x_\iota \lambda y_{\iota\to o}[y\,x]]$ is identical to $[\lambda y_\iota \lambda z_{\iota\to o}[z\,y]]$

- $[\mathbf{A}/x]\mathbf{B}$ denotes substitution of $\mathbf{A}$ for free occurrences of x in $\mathbf{B}$. We rename bound variables to ensure no capture.

Example: $[y/x][\lambda y_\iota\,[p_{\iota\to\iota\to o}\,x\,y]]$ is $[\lambda z_\iota\,[p\,y\,z]]$.

- We may also consider simultaneous substitutions θ for a finite number of variables.

More Syntax

- We will consider β and η reduction and conversion.

β: $[[\lambda \mathsf{x}_\alpha\, \mathbf{B}_\beta]\, \mathbf{A}]$ β-reduces to $[\mathbf{A}/\mathsf{x}]\mathbf{B}$

η: $[\lambda \mathsf{x}_\alpha\, [\mathbf{F}_{\alpha\to\beta}\, \mathsf{x}]]$ η-reduces to $\mathbf{F}$ if $\mathsf{x} \notin \mathbf{Free}(\mathbf{F})$

- We write $\mathbf{A} \xrightarrow{\beta}_1 \mathbf{B}$ if $\mathbf{B}$ is obtained by β-reducing in some position in $\mathbf{A}$.
- We write $\mathbf{A} \xrightarrow{\eta}_1 \mathbf{B}$ if $\mathbf{B}$ is obtained by η-reducing in some position in $\mathbf{A}$.
- We write $\xrightarrow{\beta}$ to denote the reflexive, transitive closure of $\xrightarrow{\beta}_1$.
- We write $\xrightarrow{\beta\eta}$ to denote the reflexive, transitive closure of $\xrightarrow{\beta}_1 \cup \xrightarrow{\eta}_1$.

More Syntax

- We will consider β and η reduction and conversion.

β: $[[\lambda \mathsf{x}_\alpha\, \mathbf{B}_\beta]\, \mathbf{A}]$ β-reduces to $[\mathbf{A}/\mathsf{x}]\mathbf{B}$

η: $[\lambda \mathsf{x}_\alpha\, [\mathbf{F}_{\alpha\to\beta}\, \mathsf{x}]]$ η-reduces to $\mathbf{F}$ if $\mathsf{x} \notin \mathbf{Free}(\mathbf{F})$

Facts: $\xrightarrow{\beta}$ and $\xrightarrow{\beta\eta}$ satisfy the strong Church-Rosser property: Every wff $\mathbf{A}$ has a unique normal form.

- $\mathbf{A}{\downarrow_\beta}$ denotes the β-normal (i.e., $\xrightarrow{\beta}$ normal) form of $\mathbf{A}$.
- $\mathbf{A}{\downarrow_{\beta\eta}}$ denotes the $\beta\eta$-normal (i.e., $\xrightarrow{\beta\eta}$ normal) form of $\mathbf{A}$.

Generalized Semantics

There are two key steps to generalize combinatory frames with evaluations to give nonextensional models.

To obtain non-functional semantics, we allow $\mathcal{D}_{\alpha\longrightarrow\beta}$ to be any nonempty set and include an "application operator" $@ : \mathcal{D}_{\alpha\longrightarrow\beta} \times \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$.

To generalize from two truth values, we allow $\mathcal{D}_o$ to be any nonempty set and include a "valuation" $\upsilon : \mathcal{D}_o \to \{\mathtt{T}, \mathtt{F}\}$.

Coming Attractions

1. Definition of applicative structure generalizing frames
2. Definition of logical properties relative to $\upsilon : \mathcal{D}_o \to \{\mathtt{T}, \mathtt{F}\}$.
3. Definition of evaluations for interpreting terms in applicative structures
4. Definition of model for determining which terms of type o are true
5. Definition of model classes varying extensionality

Applicative Structures

Defn: A (typed) applicative structure is a pair $(\mathcal{D}, @)$ where $\mathcal{D}$ is a typed family of nonempty sets and $@^{\alpha\to\beta} : \mathcal{D}_{\alpha\to\beta} \times \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$ for each function type $(\alpha \to \beta)$.

Write $f@a$ for $f@^{\alpha\to\beta}a$ when $f \in \mathcal{D}_{\alpha\to\beta}$ and $a \in \mathcal{D}_\alpha$ are clear in context.

Defn: Let $\mathcal{A} := (\mathcal{D}, @)$ be an applicative structure. We say $\mathcal{A}$ is functional if for all types α and β and objects $f, g \in \mathcal{D}_{\alpha\to\beta}$, $f = g$ whenever $f@a = g@a$ for every $a \in \mathcal{D}_\alpha$.

Logical Properties

Suppose $\upsilon : \mathcal{D}_o \to \{\mathtt{T}, \mathtt{F}\}$ is a function.

Defn: Let $\mathcal{A} := (\mathcal{D}, @)$ be an applicative structure and $\upsilon : \mathcal{D}_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ be a function.

For each logical constant c_α and element $a \in \mathcal{D}_\alpha$, we define the properties $\mathfrak{L}_c(a)$ with respect to υ given in the following table...

Logical Properties

prop.	where	holds when			for all
$\mathfrak{L}_\neg(n)$	$n \in \mathcal{D}_{o\to o}$	$\upsilon(n@a) = \mathtt{T}$	iff	$\upsilon(a) = \mathtt{F}$	$a \in \mathcal{D}_o$
$\mathfrak{L}_\vee(d)$	$d \in \mathcal{D}_{o\to o\to o}$	$\upsilon(d@a@b) = \mathtt{T}$	iff	$\upsilon(a) = \mathtt{T}$ or $\upsilon(b) = \mathtt{T}$	$a, b \in \mathcal{D}_o$
$\mathfrak{L}_\wedge(c)$	$c \in \mathcal{D}_{o\to o\to o}$	$\upsilon(c@a@b) = \mathtt{T}$	iff	$\upsilon(a) = \mathtt{T}$ and $\upsilon(b) = \mathtt{T}$	$a, b \in \mathcal{D}_o$
$\mathfrak{L}_{\Pi^\alpha}(\pi)$	$\pi \in \mathcal{D}_{(\alpha\to o)\to o}$	$\upsilon(\pi@f) = \mathtt{T}$	iff	$\forall a \in \mathcal{D}_\alpha\ \upsilon(f@a) = \mathtt{T}$	$f \in \mathcal{D}_{\alpha\to o}$
$\mathfrak{L}_{\Sigma^\alpha}(\sigma)$	$\sigma \in \mathcal{D}_{(\alpha\to o)\to o}$	$\upsilon(\sigma@f) = \mathtt{T}$	iff	$\exists a \in \mathcal{D}_\alpha\ \upsilon(f@a) = \mathtt{T}$	$f \in \mathcal{D}_{\alpha\to o}$
$\mathfrak{L}_{=^\alpha}(q)$	$q \in \mathcal{D}_{\alpha\to\alpha\to o}$	$\upsilon(q@a@b) = \mathtt{T}$	iff	$a = b$	$a, b \in \mathcal{D}_\alpha$

Logical Properties

Defn: Suppose $(\mathcal{D}, @)$ is an applicative structure and $\upsilon : \mathcal{D}_o \to \{\mathtt{T}, \mathtt{F}\}$ is a function.

We say $(\mathcal{D}, @, \upsilon)$ realizes a logical constant c_α if there is some $a \in \mathcal{D}_\alpha$ such that $\mathfrak{L}_c(a)$ holds with respect to this υ.
We say $(\mathcal{D}, @, \upsilon)$ realizes a signature Σ if it realizes every $c \in \Sigma$.

Variable Assignment

Defn: Let $\mathcal{A} := (\mathcal{D}, @)$ be an applicative structure.
A typed function $\varphi : \mathcal{V} \longrightarrow \mathcal{D}$ is called a variable assignment into $\mathcal{D}$.

Given a variable assignment φ, variable x_α, and value $a \in \mathcal{D}_\alpha$, we use $\varphi, [a/x]$ to denote the variable assignment with
$(\varphi, [a/x])(x) = a$ and
$(\varphi, [a/x])(y) = \varphi(y)$ for variables y other than x.

Evaluations

Defn: Let $\mathcal{A} = (\mathcal{D}, @)$ be an applicative structure.

An Σ-evaluation function $\mathcal{E}$ for $\mathcal{A}$ is a function taking assignments φ and terms $\mathbf{A}_\alpha$ to $\mathcal{E}_\varphi(\mathbf{A}) \in \mathcal{D}_\alpha$ satisfying the following properties:

1. $\mathcal{E}_\varphi(x) = \varphi(x)$ for $x \in \mathcal{V}$.
2. $\mathcal{E}_\varphi([\mathbf{F}\,\mathbf{A}]) = \mathcal{E}_\varphi(\mathbf{F})@\mathcal{E}_\varphi(\mathbf{A})$ for any $\mathbf{F} \in \mathit{wff}_{\alpha\to\beta}(\Sigma)$ and $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$ and types α and β.
3. $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\psi(\mathbf{A})$ for any type α and $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, whenever φ and ψ coincide on $\mathbf{Free}(\mathbf{A})$.
4. $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{A}{\downarrow_\beta})$ for all $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$.

Evaluations

If $\mathbf{A}$ is a closed formula, then $\mathcal{E}_\varphi(\mathbf{A})$ is independent of φ.

Then we write $\mathcal{E}(\mathbf{A})$.

Evaluations

Defn: We call $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ an Σ-evaluation if
$(\mathcal{D}, @)$ is an applicative structure and
$\mathcal{E}$ is an evaluation function for $(\mathcal{D}, @)$.

We call an Σ-evaluation $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ functional if the applicative structure $(\mathcal{D}, @)$ is functional.

We say $\mathcal{J}$ is a Σ-evaluation over a frame if $(\mathcal{D}, @)$ is a frame.

Evaluations Respect β

If A β-converts to B, then they have the same normal form.

Hence

$$\mathcal{E}_\varphi(\mathsf{A}) = \mathcal{E}_\varphi(\mathsf{A}\downarrow_\beta) = \mathcal{E}_\varphi(\mathsf{B}\downarrow_\beta) = \mathcal{E}_\varphi(\mathsf{B})$$

Substitution-Value Lemma

Substitution-Value Lemma:

$$\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B}_\beta)/\mathsf{x}_\beta]}(\mathbf{A}_\alpha) = \mathcal{E}_\varphi([\mathbf{B}/\mathsf{x}]\mathbf{A})$$

Proof:

Substitution-Value Lemma

Substitution-Value Lemma:

$$\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B}_\beta)/\mathsf{x}_\beta]}(\mathbf{A}_\alpha) = \mathcal{E}_\varphi([\mathbf{B}/\mathsf{x}]\mathbf{A})$$

Proof:

$$\begin{aligned}
\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/\mathsf{x}]}(\mathbf{A}) &= \mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/\mathsf{x}]}([[\lambda\mathsf{x}\,\mathbf{A}]\,\mathsf{x}]) \\
&= \mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/\mathsf{x}]}([\lambda\mathsf{x}\mathbf{A}])@\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/\mathsf{x}]}(\mathsf{x}) \\
&= \mathcal{E}_\varphi([\lambda\mathsf{x}\,\mathbf{A}])@\mathcal{E}_\varphi(\mathbf{B}) \\
&= \mathcal{E}_\varphi([[\lambda\mathsf{x}\,\mathbf{A}]\,\mathbf{B}]) \\
&= \mathcal{E}_\varphi([\mathbf{B}/\mathsf{x}]\mathbf{A}).
\end{aligned}$$

Substitution-Value Lemma

Substitution-Value Lemma:

$$\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B}_\beta)/\mathsf{x}_\beta]}(\mathbf{A}_\alpha) = \mathcal{E}_\varphi([\mathbf{B}/\mathsf{x}]\mathbf{A})$$

Proof:

$$\begin{aligned}
\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/\mathsf{x}]}(\mathbf{A}) &= \mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/\mathsf{x}]}([[\lambda\mathsf{x}\,\mathbf{A}]\,\mathsf{x}]) \\
&= \mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/\mathsf{x}]}([\lambda\mathsf{x}\mathbf{A}])@\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/\mathsf{x}]}(\mathsf{x}) \\
&= \mathcal{E}_\varphi([\lambda\mathsf{x}\,\mathbf{A}])@\mathcal{E}_\varphi(\mathbf{B}) \\
&= \mathcal{E}_\varphi([[\lambda\mathsf{x}\,\mathbf{A}]\,\mathbf{B}]) \\
&= \mathcal{E}_\varphi([\mathbf{B}/\mathsf{x}]\mathbf{A}).
\end{aligned}$$

Proof by Andrei Paskevich

Weak Functionality

Let $\mathcal{J} = (\mathcal{D}, @, \mathcal{E})$ be an Σ-evaluation.

We say $\mathcal{J}$ is η-functional if $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{A}{\downarrow}_{\beta\eta})$ for any type α, formula $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, and assignment φ.

We say $\mathcal{J}$ is ξ-functional if for all $\alpha, \beta \in \mathcal{T}$, $\mathbf{M}, \mathbf{N} \in \mathit{wff}_\beta(\Sigma)$, assignments φ, and variables x_α, $\mathcal{E}_\varphi([\lambda \mathsf{x}_\alpha \mathbf{M}_\beta]) = \mathcal{E}_\varphi([\lambda \mathsf{x}_\alpha \mathbf{N}_\beta])$ whenever $\mathcal{E}_{\varphi,[\mathsf{a}/\mathsf{x}]}(\mathbf{M}) = \mathcal{E}_{\varphi,[\mathsf{a}/\mathsf{x}]}(\mathbf{N})$ for every $\mathsf{a} \in \mathcal{D}_\alpha$.

$\mathfrak{f} = \eta + \xi$

functional $\Rightarrow$ η-functional

since $\mathcal{E}_\varphi([\lambda \mathsf{x}_\alpha\, [\mathbf{F}\, \mathsf{x}]]) = \mathcal{E}_\varphi(\mathbf{F})$ if $\mathsf{x} \notin \mathbf{Free}(\mathbf{F})$

functional $\Rightarrow$ ξ-functional

since $\forall \mathsf{a} \in \mathcal{D}_\alpha\, \mathcal{E}_{\varphi,[\mathsf{a},\mathsf{x}]}(\mathbf{M}) = \mathcal{E}_{\varphi,[\mathsf{a},\mathsf{x}]}(\mathbf{N}) \Rightarrow \mathcal{E}_\varphi([\lambda \mathsf{x} \mathbf{M}]) = \mathcal{E}_\varphi([\lambda \mathsf{x} \mathbf{N}])$

η-functional and ξ-functional $\Rightarrow$ functional

$$\forall \mathsf{a} \in \mathcal{D}_\alpha\; \mathsf{f}@\mathsf{a} = \mathsf{g}@\mathsf{a} \Rightarrow \mathsf{f} =_\eta \mathcal{E}_\varphi([\lambda \mathsf{x} [\mathsf{f}\, \mathsf{x}]]) =_\xi \mathcal{E}_\varphi([\lambda \mathsf{x} [\mathsf{g}\, \mathsf{x}]]) =_\eta \mathsf{g}$$

Models

Let $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ be an Σ-evaluation.

A function $\upsilon \colon \mathcal{D}_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ is called a Σ-valuation for $\mathcal{J}$ if $\mathfrak{L}_{\mathsf{c}}(\mathcal{E}(\mathsf{c}))$ holds for every $\mathsf{c} \in \Sigma$.

In this case, $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ is called an Σ-model.

Models

An assignment φ satisfies a formula $\mathbf{A} \in \mathit{wff}_o(\Sigma)$ in $\mathcal{M}$
(we write $\mathcal{M} \models_\varphi \mathbf{A}$)
if $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$.

We say that $\mathbf{A}$ is valid in $\mathcal{M}$
(and write $\mathcal{M} \models \mathbf{A}$)
if $\mathcal{M} \models_\varphi \mathbf{A}$ for all assignments φ.

When $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$, we drop φ and write $\mathcal{M} \models \mathbf{A}$.

Finally, we say that $\mathcal{M}$ is an Σ-model for a set $\Phi \subseteq \mathit{cwff}_o(\Sigma)$
(we write $\mathcal{M} \models \Phi$)

if $\mathcal{M} \models \mathbf{A}$ for all $\mathbf{A} \in \Phi$.

Example

Assume Σ contains $\neg$ and $\lor$

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model

Claim: $\mathcal{M} \models_\varphi [\lor\, P\, [\neg\, P]]$ (i.e., $P \lor \neg P$) where $P \in \mathit{wff}_o(\Sigma)$

Example

Assume Σ contains $\neg$ and $\lor$

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model

Claim: $\mathcal{M} \models_\varphi [\lor\, P\, [\neg\, P]]$ (i.e., $P \lor \neg P$) where $P \in \mathit{wff}_o(\Sigma)$

Show: $\upsilon(\mathcal{E}_\varphi([\lor\, P\, [\neg\, P]])) = \mathtt{T}$

Example

Assume Σ contains $\neg$ and $\lor$

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model

Claim: $\mathcal{M} \models_\varphi [\lor\, P\, [\neg\, P]]$ (i.e., $P \lor \neg P$) where $P \in \mathit{wff}_o(\Sigma)$

Show: $\upsilon(\mathcal{E}_\varphi([\lor\, P\, [\neg\, P]])) = \mathtt{T}$

Note: $\mathcal{E}_\varphi([\lor\, P\, [\neg\, P]]) = \mathcal{E}_\varphi(\lor)@\mathcal{E}_\varphi(P)@\mathcal{E}_\varphi([\neg\, P])$ (property of $\mathcal{E}$)

Example

Assume Σ contains $\neg$ and $\lor$

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model

Claim: $\mathcal{M} \models_\varphi [\lor\, P\, [\neg\, P]]$ (i.e., $P \lor \neg P$) where $P \in \mathit{wff}_o(\Sigma)$

Show: $\upsilon(\mathcal{E}_\varphi([\lor\, P\, [\neg\, P]])) = \mathtt{T}$

Note: $\mathcal{E}_\varphi([\lor\, P\, [\neg\, P]]) = \mathcal{E}_\varphi(\lor)@\mathcal{E}_\varphi(P)@\mathcal{E}_\varphi([\neg\, P])$ (property of $\mathcal{E}$)

Use $\mathfrak{L}_\lor(\mathcal{E}(\lor))$ – Show: Either $\upsilon(\mathcal{E}_\varphi(P)) = \mathtt{T}$ or $\upsilon(\underbrace{\mathcal{E}_\varphi([\neg\, P])}_{\mathcal{E}_\varphi(\neg)@\mathcal{E}_\varphi(P)}) = \mathtt{T}$

Example

Assume Σ contains $\neg$ and $\vee$

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model

Claim: $\mathcal{M} \models_\varphi [\vee\, \mathsf{P}\, [\neg\, \mathsf{P}]]$ (i.e., $\mathsf{P} \vee \neg\, \mathsf{P}$) where $\mathsf{P} \in wff_o(\Sigma)$

Show: $\upsilon(\mathcal{E}_\varphi([\vee\, \mathsf{P}\, [\neg\, \mathsf{P}]])) = \mathtt{T}$

Note: $\mathcal{E}_\varphi([\vee\, \mathsf{P}\, [\neg\, \mathsf{P}]]) = \mathcal{E}_\varphi(\vee) @ \mathcal{E}_\varphi(\mathsf{P}) @ \mathcal{E}_\varphi([\neg\, \mathsf{P}])$ (property of $\mathcal{E}$)

Use $\mathfrak{L}_\vee(\mathcal{E}(\vee))$ – Show: Either $\upsilon(\mathcal{E}_\varphi(\mathsf{P})) = \mathtt{T}$ or $\upsilon(\underbrace{\mathcal{E}_\varphi([\neg\, \mathsf{P}])}_{\mathcal{E}_\varphi(\neg) @ \mathcal{E}_\varphi(\mathsf{P})}) = \mathtt{T}$

Use $\mathfrak{L}_\neg(\mathcal{E}(\neg))$ – Show: Either $\upsilon(\mathcal{E}_\varphi(\mathsf{P})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_\varphi(\mathsf{P})) = \mathtt{F}$

Example

Assume Σ contains $\neg$ and $\vee$

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model

Claim: $\mathcal{M} \models_\varphi [\vee\, \mathsf{P}\, [\neg\, \mathsf{P}]]$ (i.e., $\mathsf{P} \vee \neg\, \mathsf{P}$) where $\mathsf{P} \in wff_o(\Sigma)$

Show: $\upsilon(\mathcal{E}_\varphi([\vee\, \mathsf{P}\, [\neg\, \mathsf{P}]])) = \mathtt{T}$

Note: $\mathcal{E}_\varphi([\vee\, \mathsf{P}\, [\neg\, \mathsf{P}]]) = \mathcal{E}_\varphi(\vee) @ \mathcal{E}_\varphi(\mathsf{P}) @ \mathcal{E}_\varphi([\neg\, \mathsf{P}])$ (property of $\mathcal{E}$)

Use $\mathfrak{L}_\vee(\mathcal{E}(\vee))$ – Show: Either $\upsilon(\mathcal{E}_\varphi(\mathsf{P})) = \mathtt{T}$ or $\upsilon(\underbrace{\mathcal{E}_\varphi([\neg\, \mathsf{P}])}_{\mathcal{E}_\varphi(\neg) @ \mathcal{E}_\varphi(\mathsf{P})}) = \mathtt{T}$

Use $\mathfrak{L}_\neg(\mathcal{E}(\neg))$ – Show: Either $\upsilon(\mathcal{E}_\varphi(\mathsf{P})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_\varphi(\mathsf{P})) = \mathtt{F}$

OK, since $\upsilon : \mathcal{D}_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$.

Properties of Models

A Σ-model $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ is called functional if the applicative structure $(\mathcal{D}, @)$ is functional.

Similarly, $\mathcal{M}$ is called η-functional [ξ-functional] if the evaluation $(\mathcal{D}, @, \mathcal{E})$ is η-functional [ξ-functional].

We say $\mathcal{M}$ is an Σ-model over a frame if $(\mathcal{D}, @)$ is a frame.

Properties

Given an Σ-model $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$, we say that $\mathcal{M}$ has property

- $\mathfrak{q}$ iff for all $\alpha \in \mathcal{T}$ there is some $\mathsf{q}^\alpha \in \mathcal{D}_{\alpha \to \alpha \to o}$ such that $\mathfrak{L}_{=^\alpha}(\mathsf{q}^\alpha)$ holds.
- η iff $\mathcal{M}$ is η-functional.
- ξ iff $\mathcal{M}$ is ξ-functional.
- $\mathfrak{f}$ iff $\mathcal{M}$ is functional. (This is generally associated with functional extensionality.)
- $\mathfrak{b}$ iff υ is injective (and so $\mathcal{D}_o$ has at most two elements).

Signature Restriction

NOTE: From now on, we restrict to the signature Σ being either $\boxed{\{\neg, \vee\} \cup \{\Pi^\alpha | \alpha \in \mathcal{T}\}}$ or $\boxed{\{\neg, \vee\} \cup \{\Pi^\alpha, =^\alpha | \alpha \in \mathcal{T}\}}$.
Unless otherwise noted, other logical "constants" are abbreviations:

- $\supset$ is $[\lambda p_o\, \lambda q_o\, [\neg p \vee q]]$
- $\wedge$ is $[\lambda p_o\, \lambda q_o\, \neg[\neg p \vee \neg q]]$
- $\Leftrightarrow$ is $[\lambda p_o\, \lambda q_o\, [[p \supset q] \wedge [q \supset p]]]$
- Σ^α is $[\lambda p_{\alpha\to o}\, \neg[\Pi^\alpha\, [\lambda x_\alpha\, \neg[p\, x]]]]$

We sometimes consider "Leibniz Equality" denoted $\doteq^\alpha$:

$$[\lambda x_\alpha \lambda y_\alpha \forall p_{\alpha\to o}[[p\, x] \supset [p\, y]]]$$

Model Classes

Denote class of Σ-models that satisfy property $\mathfrak{q}$ by $\mathfrak{M}_\beta(\Sigma)$.

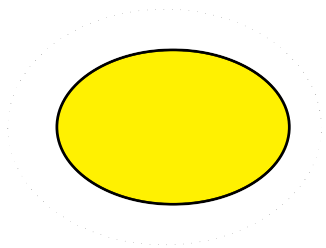
Specialized subclasses of depending on the validity of the properties η, ξ, $\mathfrak{f}$, and $\mathfrak{b}$ denoted by

$\mathfrak{M}_{\beta\eta}(\Sigma)$, $\mathfrak{M}_{\beta\xi}(\Sigma)$, $\mathfrak{M}_{\beta\mathfrak{f}}(\Sigma)$, $\mathfrak{M}_{\beta\mathfrak{b}}(\Sigma)$,

$\mathfrak{M}_{\beta\eta\mathfrak{b}}(\Sigma)$, $\mathfrak{M}_{\beta\xi\mathfrak{b}}(\Sigma)$, and $\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}(\Sigma)$.

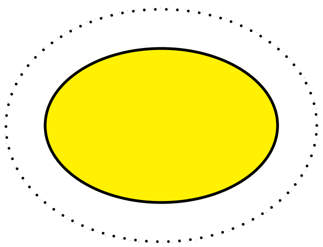
Semantics: HOL-CUBE

Standard Models and Henkin Models



Standard Models $\mathfrak{S}\mathfrak{T}(\Sigma)$

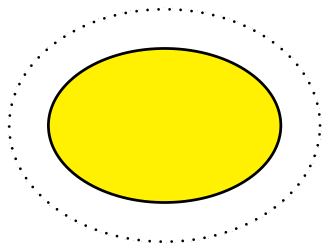
Standard Models and Henkin Models



Standard Models $\mathfrak{ST}(\Sigma)$

- Leon Henkin [Henkin50] generalized the class of admissible domains for functional types.

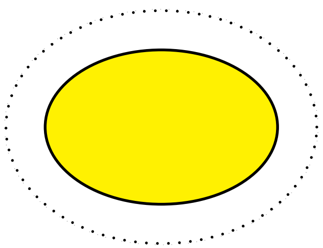
Standard Models and Henkin Models



Standard Models $\mathfrak{ST}(\Sigma)$

- Leon Henkin [Henkin50] generalized the class of admissible domains for functional types.
- Instead of requiring $\mathcal{D}_{\alpha\to\beta}$ (and thus in particular, $\mathcal{D}_{\iota\to o}$) to be the full set of functions (predicates), it is sufficient to require that $\mathcal{D}_{\alpha\to\beta}$ has enough members that any well-formed formula can be evaluated.

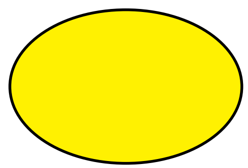
Standard Models and Henkin Models



Standard Models $\mathfrak{ST}(\Sigma)$

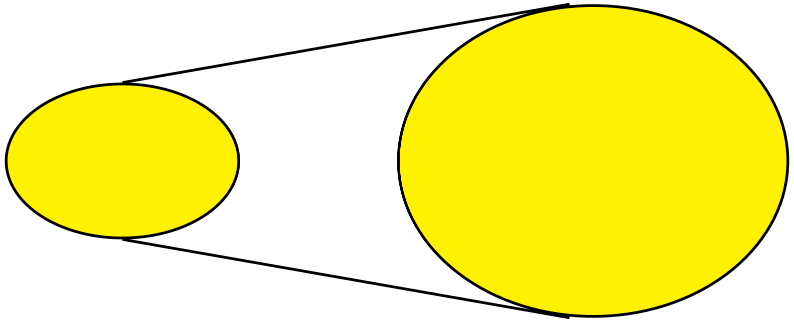
- Leon Henkin [Henkin50] generalized the class of admissible domains for functional types.
- Instead of requiring $\mathcal{D}_{\alpha\to\beta}$ (and thus in particular, $\mathcal{D}_{\iota\to o}$) to be the full set of functions (predicates), it is sufficient to require that $\mathcal{D}_{\alpha\to\beta}$ has enough members that any well-formed formula can be evaluated.
- Note that with this generalized notion of a model, there are fewer formulae that are valid in all models (intuitively, for any given formula there are more possibilities for counter-models).

HOL Models and Valid Formulas



Standard Models $\mathfrak{ST}(\Sigma)$

HOL Models and Valid Formulas

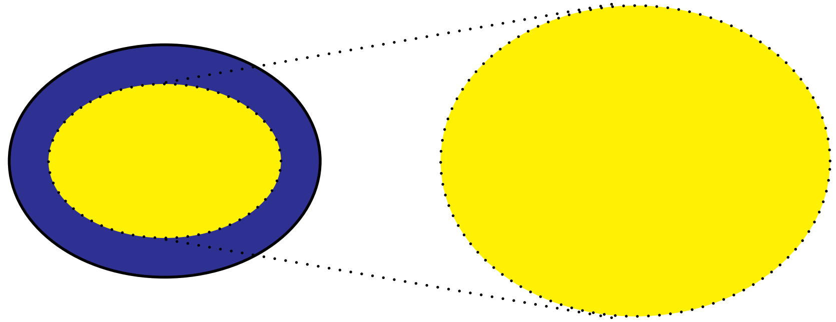


Standard Models $\mathfrak{ST}(\Sigma)$

Formulas valid in $\mathfrak{ST}(\Sigma)$

Ex.: Cantor's Theorem

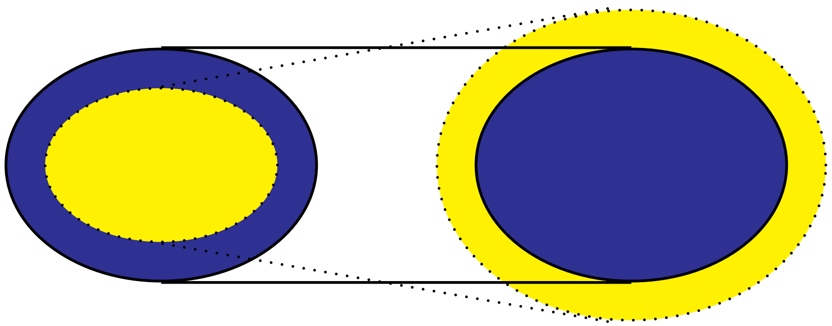
HOL Models and Valid Formulas



Henkin Models $\mathfrak{H}(\Sigma) = \mathfrak{M}_{\beta\mathfrak{fb}}(\Sigma)$

Formulas valid in $\mathfrak{M}_{\beta\mathfrak{fb}}(\Sigma)$?

HOL Models and Valid Formulas

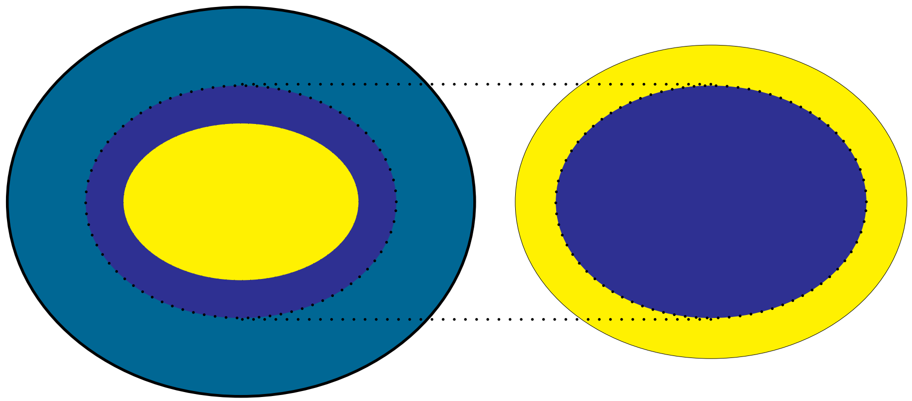


Henkin Models $\mathfrak{H}(\Sigma) = \mathfrak{M}_{\beta\mathfrak{fb}}(\Sigma)$

Formulas valid in $\mathfrak{M}_{\beta\mathfrak{fb}}(\Sigma)$

Ex.: Cantor's Theorem (for certain Σ)
not valid anymore

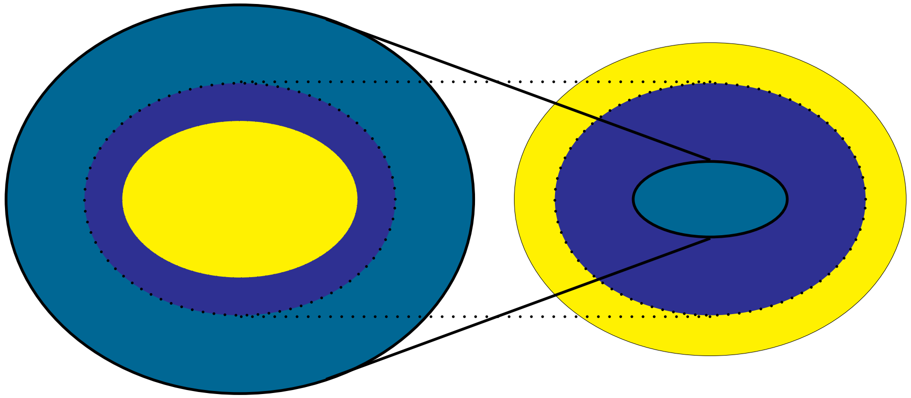
HOL Models and Valid Formulas



Non-Extensional Models $\mathfrak{M}_{\beta}(\Sigma)$

Formulas valid in $\mathfrak{M}_{\beta}(\Sigma)$?

HOL Models and Valid Formulas

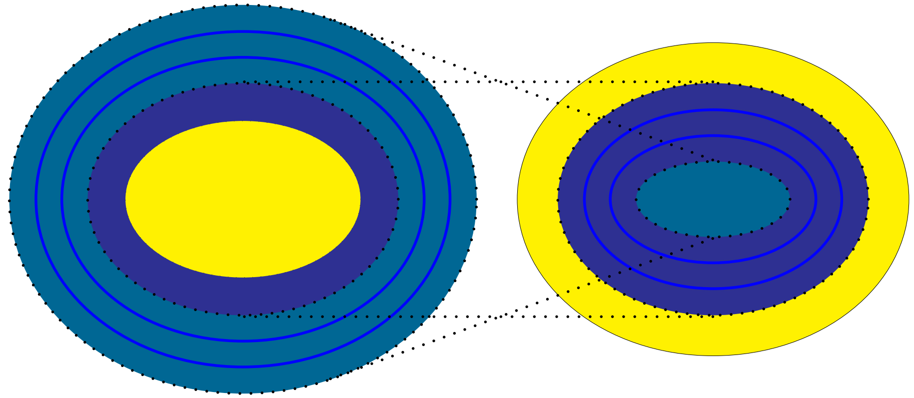


Non-Extensional Models $\mathfrak{M}_\beta(\Sigma)$

Formulas valid in $\mathfrak{M}_\beta(\Sigma)$?

Ex.: $\forall X. \forall Y. X \vee Y \Leftrightarrow Y \vee X$
vs. $\vee \doteq \lambda X. \lambda Y. Y \vee X$

HOL Models and Valid Formulas



We additionally studied different model classes with 'varying degrees of extensionality'

$\forall X. \forall Y. X \vee Y \Leftrightarrow Y \vee X$ $\quad$ $\forall X. \forall Y. X \vee Y \doteq Y \vee X$

$\lambda X. \lambda Y. X \vee Y \doteq \lambda X. \lambda Y. Y \vee X$ $\quad$ $\vee \doteq \lambda X. \lambda Y. Y \vee X$

Primitive and Defined Equality

Some conventions:

- $=$ denotes primitive equality

Primitive and Defined Equality

Some conventions:

- $=$ denotes primitive equality
- $\doteq$ denotes Leibniz equality:

$$\mathbf{A}_\alpha \doteq^\alpha \mathbf{B}_\alpha := \forall P_{o\alpha}. (P\mathbf{A}) \Rightarrow (P\mathbf{B})$$

Primitive and Defined Equality

Some conventions:

- $=$ denotes primitive equality
- $\doteq$ denotes Leibniz equality:

$$\mathbf{A}_\alpha \doteq^\alpha \mathbf{B}_\alpha := \forall P_{o\alpha}.(P\mathbf{A}) \Rightarrow (P\mathbf{B})$$

$$\doteq^\alpha \; := \lambda X_\alpha.\lambda Y_\alpha.\Pi^\alpha(\lambda P_{\alpha\to o}.\neg(PX) \vee (PY))$$

Primitive and Defined Equality

Some conventions:

- $=$ denotes primitive equality
- $\doteq$ denotes Leibniz equality:

$$\mathbf{A}_\alpha \doteq^\alpha \mathbf{B}_\alpha := \forall P_{o\alpha}.(P\mathbf{A}) \Rightarrow (P\mathbf{B})$$

$$\doteq^\alpha \; := \lambda X_\alpha.\lambda Y_\alpha.\Pi^\alpha(\lambda P_{\alpha\to o}.\neg(PX) \vee (PY))$$

- $\ddot{=}$ ...other definition of equality (e.g., see [[Andrews02]])

Primitive and Defined Equality

Some conventions:

- $=$ denotes primitive equality
- $\doteq$ denotes Leibniz equality:

$$\mathbf{A}_\alpha \doteq^\alpha \mathbf{B}_\alpha := \forall P_{o\alpha}.(P\mathbf{A}) \Rightarrow (P\mathbf{B})$$

$$\doteq^\alpha \; := \lambda X_\alpha.\lambda Y_\alpha.\Pi^\alpha(\lambda P_{\alpha\to o}.\neg(PX) \vee (PY))$$

- $\ddot{=}$ ...other definition of equality (e.g., see [[Andrews02]])

We use $\stackrel{*}{=}$ in the following to refer to any of the above

Semantics: HOL-CUBE

$\mathfrak{M}_\beta(\Sigma)$

$\mathfrak{b}$: υ is injective ($|\mathcal{D}_o| = 2$)

$\mathfrak{f}(= \eta + \xi)$: $\mathcal{M}$ is functional

η: $\mathcal{M}$ is η-functional

ξ: $\mathcal{M}$ is ξ-functional

$\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}(\Sigma) \simeq \mathfrak{H}(\Sigma)$

full

$\mathfrak{ST}(\Sigma)$

$\mathfrak{M}_\beta(\Sigma)$ elementary type theory (Σ-models)

Assume that logical symbols are $\{\neg, \vee\} \cup \{\Pi^\alpha\}$ or $\{\neg, \vee\} \cup \{\Pi^\alpha, =^\alpha\}$

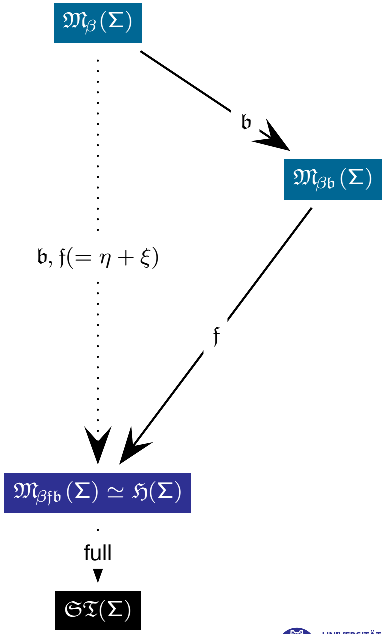
We also require property $\mathfrak{q}$:

$\forall\alpha : \mathrm{id} \in \mathcal{D}_{\alpha\to\alpha\to o}$

without this equality $\doteq$ not necessarily evaluates to identity relation even in Henkin models [[Andrews72]]

$\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}(\Sigma) \simeq \mathfrak{H}(\Sigma)$ extensional type theory (Henkin models)

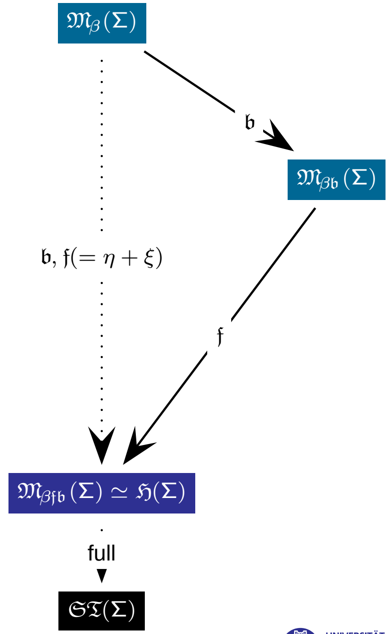
Semantics: HOL-CUBE



Motivation for Models without Functional Extensionality

- modeling programs: $p_1 \neq p_2$ even if $f@a = g@a$ for every $a \in \mathcal{D}_\alpha$
- consider properties like run-time complexity:
- $\mathbf{P}_1 := \lambda X_{nat}.1$ and $\mathbf{P}_2 := \lambda X_{nat}.1 + (X+1)^2 - (X^2 + 2X + 1)$
- $\mathbf{P}_1$ has constant complexity, $\mathbf{P}_2$ has not
- however, $\mathbf{P}_1$ behaves like $\mathbf{P}_2$ on all inputs
- a logic with a functionally extensional model theory (property $\mathfrak{f}$) necessarily conflates $\mathbf{P}_1$ and $\mathbf{P}_2$ semantically

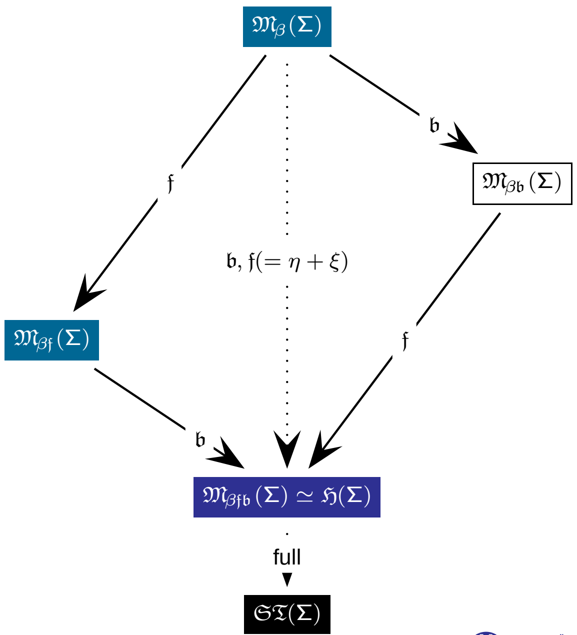
Semantics: HOL-CUBE



How do we account for Models without Functional Extensionality?

- generalized the notion of domains at function types and evaluation functions
- example: $(\mathsf{efficient}, K_1) \neq (\mathsf{inefficient}, K_1) \in \mathcal{D}_{nat \to nat}$ where K_1 is the constant-1 function and $(*^1, *^2)@n$ is defined as $*^2(n)$
- we build on the notion of applicative structures to define Σ-evaluations, where the evaluation function is assumed to respect application and β-conversion

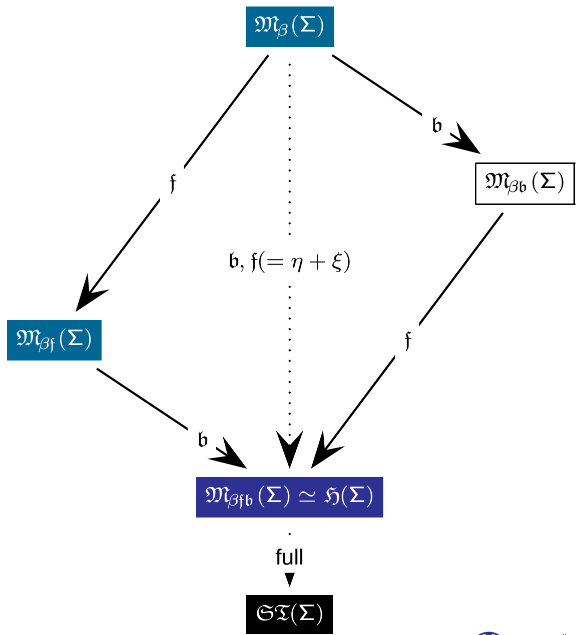
Semantics: HOL-CUBE



Motivation for models without Boolean Extensionality?

- modeling of intensional concepts like 'knowledge', 'believe', etc.
- example:
  $\mathbf{O} := 2 + 2 = 4$
  $\mathbf{F} := \forall x, y, z, n > 2.x^n + y^n = z^n \Rightarrow x = y = z = 0$
  We want to model:
  (1) $\mathbf{O} \Leftrightarrow \mathbf{F}$ is true
  (2)
  $\mathsf{john\_knows}(\mathbf{F}) \not\Leftrightarrow \mathsf{john\_knows}(\mathbf{O})$
- if we have $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ then
  (1) implies $\mathbf{O} = \mathbf{F}$
  which enforces
  $\mathsf{john\_knows}(\mathbf{F}) = \mathsf{john\_knows}(\mathbf{O})$
  and
  $\mathsf{john\_knows}(\mathbf{F}) \Leftrightarrow \mathsf{john\_knows}(\mathbf{O})$

Semantics: HOL-CUBE



How do we account for models without Boolean Extensionality?

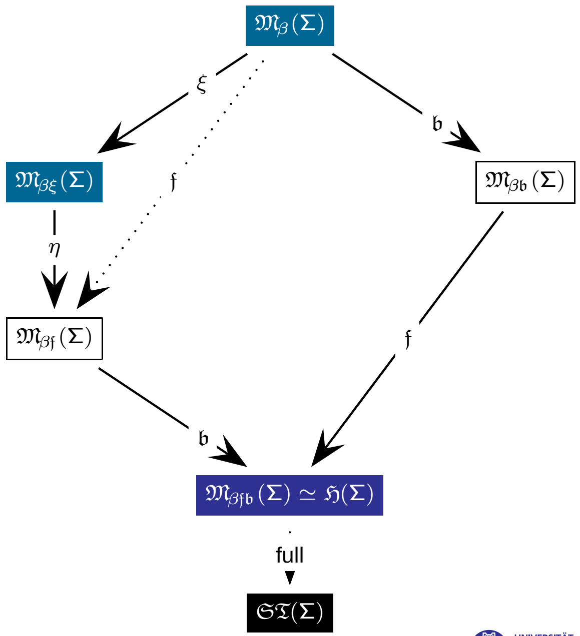
- allow that $|\mathcal{D}_o| > 2$ and use υ
- partition $|\mathcal{D}_o|$ into representatives of $\mathtt{T}$ and $\mathtt{F}$;
  e.g. $\mathcal{D}_o := \{\bot^1, \bot^2, \top^1, \top^2\}$ with $\upsilon(\bot^*) = \mathtt{F}$ and $\upsilon(\top^*) = \mathtt{T}$
- now, a predicate like $\mathsf{john\_knows}$ may map:

  $\top^1 \longrightarrow \top^1$
  $\top^2 \longrightarrow \bot^1$
  $\bot^1 \longrightarrow \bot^1$
  $\bot^2 \longrightarrow \top^1$

  and we may choose:

  $\mathbf{O}$ evaluates to $\top^1$
  $\mathbf{F}$ evaluates to $\top^2$

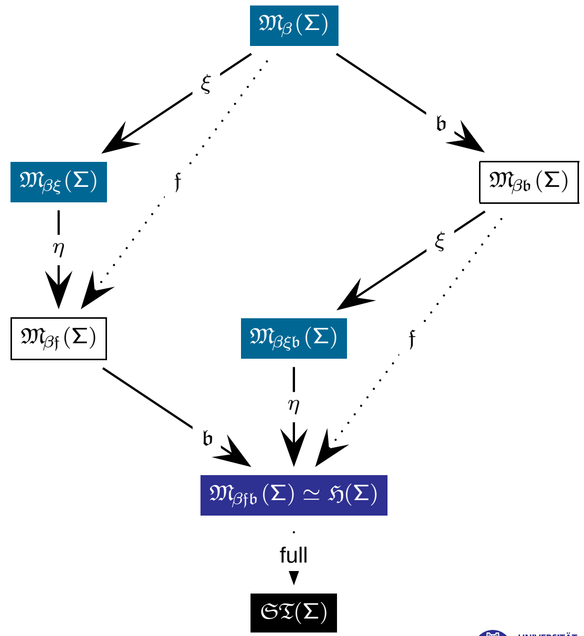
Semantics: HOL-CUBE



Models without η

$$\mathcal{E}_\varphi(A) = \mathcal{E}_\varphi(A\downarrow_{\beta\eta})$$

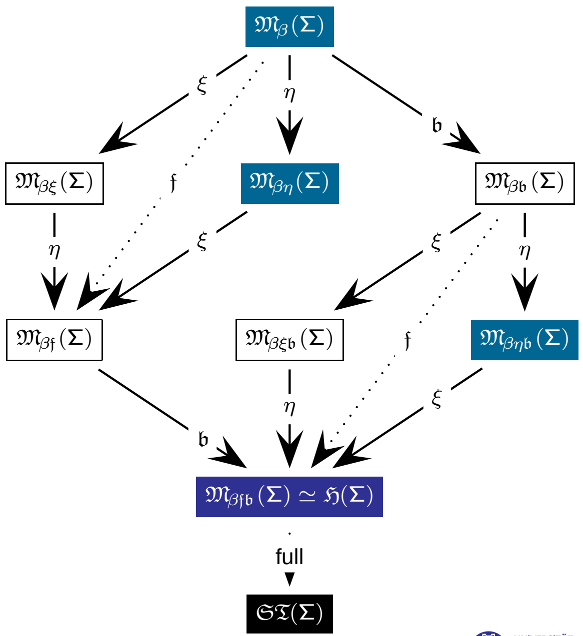
Semantics: HOL-CUBE



Models without η

$$\mathcal{E}_\varphi(A) = \mathcal{E}_\varphi(A\downarrow_{\beta\eta})$$

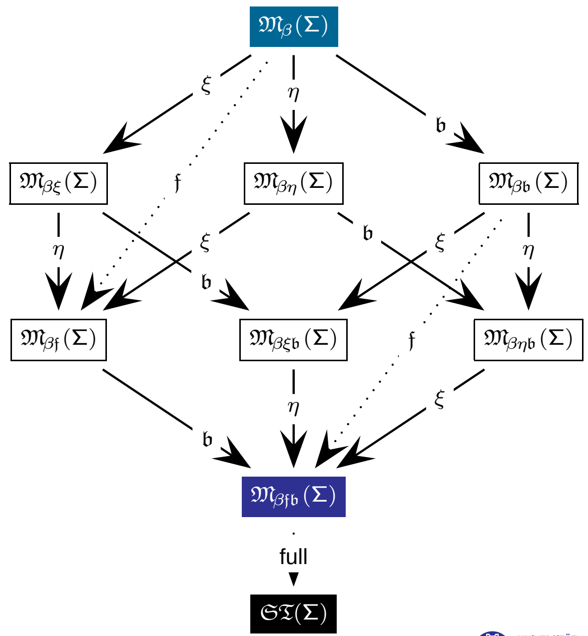
Semantics: HOL-CUBE



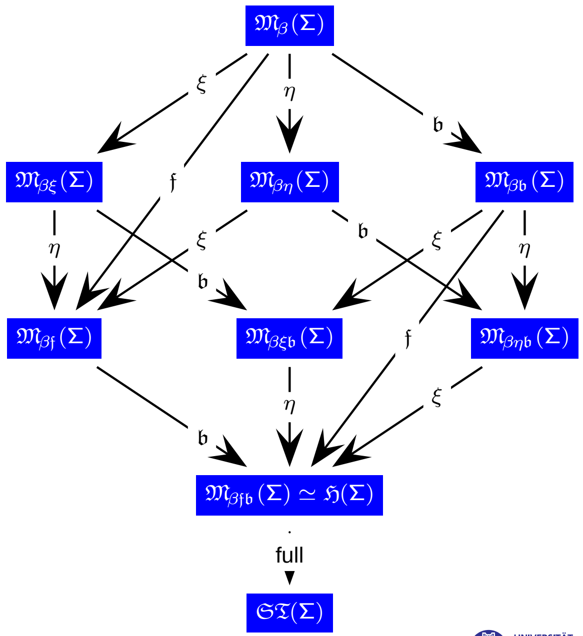
Models without ξ

$$\mathcal{E}_\varphi(\lambda X_\alpha.M_\beta) = \mathcal{E}_\varphi(\lambda X_\alpha.N_\beta) \text{ iff } \mathcal{E}_{\varphi,[a/X]}(M) = \mathcal{E}_{\varphi,[a/X]}(N)\ (\forall a \in \mathcal{D}_\alpha)$$

Semantics: HOL-CUBE



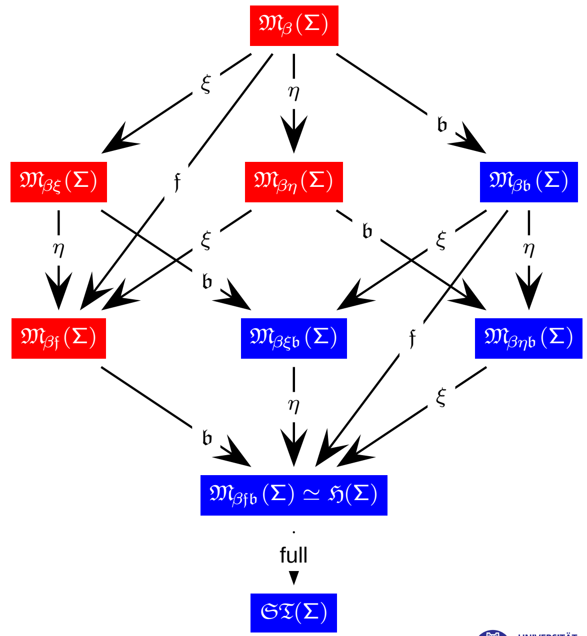
HOL Example Problems



- $\forall X. \forall Y. X \lor Y \Leftrightarrow Y \lor X$

valid for all model classes

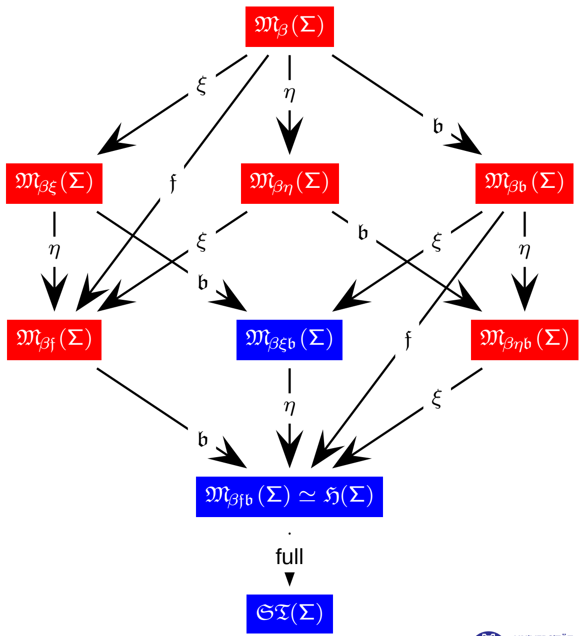
HOL Example Problems



- $\forall X. \forall Y. X \lor Y \Leftrightarrow Y \lor X$
- $\forall X. \forall Y. X \lor Y \doteq Y \lor X$

validity requires $\mathfrak{b}$

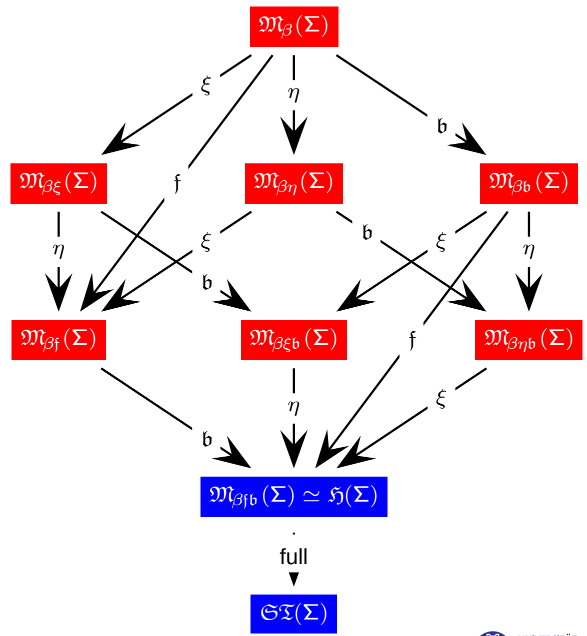
HOL Example Problems



- $\forall X. \forall Y. X \lor Y \Leftrightarrow Y \lor X$
- $\forall X. \forall Y. X \lor Y \doteq Y \lor X$
- $\lambda X. \lambda Y. X \lor Y \doteq \lambda X. \lambda Y. Y \lor X$

validity requires $\mathfrak{b}$ and ξ

HOL Example Problems



- $\forall X. \forall Y. X \lor Y \Leftrightarrow Y \lor X$
- $\forall X. \forall Y. X \lor Y \doteq Y \lor X$
- $\lambda X. \lambda Y. X \lor Y \doteq \lambda X. \lambda Y. Y \lor X$
- $\lor \doteq \lambda X. \lambda Y. Y \lor X$

validity requires $\mathfrak{b}$ and $\mathfrak{f}$

Defined Logical Connectives
in Σ-Models

Defined Logical Connectives

Lemma: (Truth and Falsity in Σ-Models)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model and φ an assignment.
Let $\mathbf{T}_o := \forall P_o. P \vee \neg P$ and $\mathbf{F}_o := \neg \mathbf{T}_o$.
Then $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{F}_o)) = \mathtt{F}$.

Defined Logical Connectives

Lemma: (Truth and Falsity in Σ-Models)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model and φ an assignment.
Let $\mathbf{T}_o := \forall P_o. P \vee \neg P$ and $\mathbf{F}_o := \neg \mathbf{T}_o$.
Then $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{F}_o)) = \mathtt{F}$.

Defined Logical Connectives

Lemma: (Truth and Falsity in Σ-Models)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model and φ an assignment.
Let $\mathbf{T}_o := \forall P_o. P \vee \neg P$ and $\mathbf{F}_o := \neg \mathbf{T}_o$.
Then $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{F}_o)) = \mathtt{F}$.

Proof: $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$

Defined Logical Connectives

Lemma: (Truth and Falsity in Σ-Models)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model and φ an assignment.
Let $\mathbf{T}_o := \forall \mathsf{P}_o \mathsf{.P} \lor \neg \mathsf{P}$ and $\mathbf{F}_o := \neg \mathbf{T}_o$.
Then $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{F}_o)) = \mathtt{F}$.

Proof: $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$

- iff $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P} \lor \neg \mathsf{P})) = \mathtt{T}$ for all $\mathsf{p} \in \mathcal{D}_o$

Defined Logical Connectives

Lemma: (Truth and Falsity in Σ-Models)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model and φ an assignment.
Let $\mathbf{T}_o := \forall \mathsf{P}_o \mathsf{.P} \lor \neg \mathsf{P}$ and $\mathbf{F}_o := \neg \mathbf{T}_o$.
Then $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{F}_o)) = \mathtt{F}$.

Proof: $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$

- iff $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P} \lor \neg \mathsf{P})) = \mathtt{T}$ for all $\mathsf{p} \in \mathcal{D}_o$
- This is equivalent to $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P})) = \mathtt{F}$.

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[2] – p.135

Defined Logical Connectives

Lemma: (Truth and Falsity in Σ-Models)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model and φ an assignment.
Let $\mathbf{T}_o := \forall \mathsf{P}_o \mathsf{.P} \lor \neg \mathsf{P}$ and $\mathbf{F}_o := \neg \mathbf{T}_o$.
Then $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{F}_o)) = \mathtt{F}$.

Proof: $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$

- iff $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P} \lor \neg \mathsf{P})) = \mathtt{T}$ for all $\mathsf{p} \in \mathcal{D}_o$
- This is equivalent to $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P})) = \mathtt{F}$.
- This is equivalent to $\upsilon(\varphi[\mathsf{p}/\mathsf{P}](\mathsf{P})) = \mathtt{T}$ or $\upsilon(\varphi[\mathsf{p}/\mathsf{P}](\mathsf{P})) = \mathtt{F}$.

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[2] – p.135

Defined Logical Connectives

Lemma: (Truth and Falsity in Σ-Models)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model and φ an assignment.
Let $\mathbf{T}_o := \forall \mathsf{P}_o \mathsf{.P} \lor \neg \mathsf{P}$ and $\mathbf{F}_o := \neg \mathbf{T}_o$.
Then $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{F}_o)) = \mathtt{F}$.

Proof: $\upsilon(\mathcal{E}_\varphi(\mathbf{T}_o)) = \mathtt{T}$

- iff $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P} \lor \neg \mathsf{P})) = \mathtt{T}$ for all $\mathsf{p} \in \mathcal{D}_o$
- This is equivalent to $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_{\varphi[\mathsf{p}/\mathsf{P}]}(\mathsf{P})) = \mathtt{F}$.
- This is equivalent to $\upsilon(\varphi[\mathsf{p}/\mathsf{P}](\mathsf{P})) = \mathtt{T}$ or $\upsilon(\varphi[\mathsf{p}/\mathsf{P}](\mathsf{P})) = \mathtt{F}$.
- Since υ maps into $\{\mathtt{T}, \mathtt{F}\}$ this must be true.

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[2] – p.135

Defined Logical Connectives

Rem.: ($|\mathcal{D}_o| \geq 2$ and υ surjective)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model. By the previous Lemma, $\mathcal{D}_o$ must have at least the two elements $\mathcal{E}_\varphi(\mathbf{T}_o)$ and $\mathcal{E}_\varphi(\mathbf{F}_o)$, and υ must be surjective.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof:

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof: Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof: Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$.

- This implies $\upsilon(\mathcal{E}_\varphi(\neg(\neg(\neg\mathbf{A} \vee \mathbf{B}) \vee \neg(\neg\mathbf{B} \vee \mathbf{A})))) = \mathtt{T}$

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof: Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$.

- This implies $\upsilon(\mathcal{E}_\varphi(\neg(\neg(\neg\mathbf{A} \vee \mathbf{B}) \vee \neg(\neg\mathbf{B} \vee \mathbf{A})))) = \mathtt{T}$
- This implies $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) = \mathtt{T}$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof: Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$.

- This implies $\upsilon(\mathcal{E}_\varphi(\neg(\neg(\neg\mathbf{A} \vee \mathbf{B}) \vee \neg(\neg\mathbf{B} \vee \mathbf{A})))) = \mathtt{T}$
- This implies $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) = \mathtt{T}$.
- If $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$, then $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) = \mathtt{T}$ implies $\upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$, so $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T} = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof: Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$.

- This implies $\upsilon(\mathcal{E}_\varphi(\neg(\neg(\neg\mathbf{A} \vee \mathbf{B}) \vee \neg(\neg\mathbf{B} \vee \mathbf{A})))) = \mathtt{T}$
- This implies $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) = \mathtt{T}$.
- If $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$, then $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) = \mathtt{T}$ implies $\upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$, so $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T} = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.
- If $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{F}$, then $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) = \mathtt{T}$ implies $\upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$, so $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{F} = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof: Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$.

- This implies $\upsilon(\mathcal{E}_\varphi(\neg(\neg(\neg\mathbf{A} \vee \mathbf{B}) \vee \neg(\neg\mathbf{B} \vee \mathbf{A})))) = \mathtt{T}$
- This implies $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) = \mathtt{T}$.
- If $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$, then $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) = \mathtt{T}$ implies $\upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$, so $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T} = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.
- If $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{F}$, then $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) = \mathtt{T}$ implies $\upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$, so $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{F} = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.
- Since these are the only two possible values for $\upsilon(\mathcal{E}_\varphi(\mathbf{A}))$, we have $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof:

Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof:

Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

- Either $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof:

Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

- Either $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$.
- An easy consideration of both cases verifies $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) = \mathtt{T}$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof:

Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

- Either $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$.
- An easy consideration of both cases verifies $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \lor \mathbf{B})) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \lor \mathbf{A})) = \mathtt{T}$.
- Hence, $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof:

Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

- Either $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$.
- An easy consideration of both cases verifies $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \lor \mathbf{B})) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \lor \mathbf{A})) = \mathtt{T}$.
- Hence, $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$.

q.e.d.

Defined Logical Connectives

Lemma: (Equivalence)
Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in wff_o(\Sigma)$.
$\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$ iff $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

Proof:

Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B}))$.

- Either $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$ or $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$.
- An easy consideration of both cases verifies $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{A} \lor \mathbf{B})) = \mathtt{T}$ and $\upsilon(\mathcal{E}_\varphi(\neg\mathbf{B} \lor \mathbf{A})) = \mathtt{T}$.
- Hence, $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) = \mathtt{T}$.

q.e.d.

Extensionality for Leibniz Equality

Def.: (Extensionality for Leibniz Equality)

Extensionality for Leibniz Equality

Def.: (Extensionality for Leibniz Equality)

We call a formula of the form

$$\mathsf{EXT}_{\dot{=}}^{\alpha\to\beta} \quad := \quad \forall F_{\alpha\to\beta}.\forall G_{\alpha\to\beta}.(\forall X_\alpha.FX \doteq^\beta GX) \Rightarrow F \doteq^{\alpha\to\beta} G$$

an axiom of (strong) functional extensionality for Leibniz equality.

Extensionality for Leibniz Equality

Def.: (Extensionality for Leibniz Equality)

We call a formula of the form

$$\mathsf{EXT}_{\dot{=}}^{\alpha\to\beta} \quad := \quad \forall F_{\alpha\to\beta}.\forall G_{\alpha\to\beta}.(\forall X_\alpha.FX \doteq^\beta GX) \Rightarrow F \doteq^{\alpha\to\beta} G$$

an axiom of (strong) functional extensionality for Leibniz equality.

We refer to the set

$$\mathsf{EXT}_{\dot{=}}^{\to} := \{\mathsf{EXT}_{\dot{=}}^{\alpha\to\beta} \mid \alpha, \beta \in \mathcal{T}\}$$

as the axioms of (strong) functional extensionality for Leibniz equality.

Extensionality for Leibniz Equality

Def.: (Extensionality for Leibniz Equality)

Extensionality for Leibniz Equality

Def.: (Extensionality for Leibniz Equality)

We call the formula

$$\mathsf{EXT}_{\dot{=}}^{o} \quad := \quad \forall A_o.\forall B_o.(A \Leftrightarrow B) \Rightarrow A \doteq^o B$$

the axiom of Boolean extensionality.

Extensionality for Leibniz Equality

Def.: (Extensionality for Leibniz Equality)

We call the formula

$$\mathsf{EXT}^{o}_{\doteq} \quad := \quad \forall A_o . \forall B_o . (A \Leftrightarrow B) \Rightarrow A \doteq^{o} B$$

the axiom of Boolean extensionality.

We call the set $\mathsf{EXT}^{\rightarrow}_{\doteq} \cup \{\mathsf{EXT}^{o}_{\doteq}\}$ the axioms of (strong) extensionality for Leibniz equality.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_{\alpha}(\Sigma)$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_{\alpha}(\Sigma)$.

1. If $\mathcal{E}_{\varphi}(\mathbf{A}) = \mathcal{E}_{\varphi}(\mathbf{B})$, then $\upsilon(\mathcal{E}_{\varphi}(\mathbf{A} \doteq^{\alpha} \mathbf{B})) = \mathtt{T}$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_{\alpha}(\Sigma)$.

1. If $\mathcal{E}_{\varphi}(\mathbf{A}) = \mathcal{E}_{\varphi}(\mathbf{B})$, then $\upsilon(\mathcal{E}_{\varphi}(\mathbf{A} \doteq^{\alpha} \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_{\varphi}(\mathbf{A} \doteq^{\alpha} \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_{\varphi}(\mathbf{A}) = \mathcal{E}_{\varphi}(\mathbf{B})$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof:

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof:

Let φ be any assignment into $\mathcal{M}$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof:

Let φ be any assignment into $\mathcal{M}$.

- For the first part, suppose $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof:

Let φ be any assignment into $\mathcal{M}$.

- For the first part, suppose $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.
- Given $\mathsf{r} \in \mathcal{D}_{\alpha\to o}$, we have either $\upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$ or $\upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{B})) = \upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof:

Let φ be any assignment into $\mathcal{M}$.

- For the first part, suppose $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.
- Given $\mathsf{r} \in \mathcal{D}_{\alpha\to o}$, we have either $\upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$ or $\upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{B})) = \upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$.
- In either case, for any variable $\mathsf{P}_{\alpha\to o}$ not in $\mathbf{Free}(\mathbf{A}) \cup \mathbf{Free}(\mathbf{B})$, we have $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\neg(\mathsf{P}\mathbf{A}) \vee \mathsf{P}\mathbf{B})) = \mathtt{T}$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof:

Let φ be any assignment into $\mathcal{M}$.

- For the first part, suppose $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.
- Given $\mathsf{r} \in \mathcal{D}_{\alpha\to o}$, we have either $\upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{A})) = \upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{F}$ or $\upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{B})) = \upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$.
- In either case, for any variable $\mathsf{P}_{\alpha\to o}$ not in $\mathbf{Free}(\mathbf{A}) \cup \mathbf{Free}(\mathbf{B})$, we have $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\neg(\mathsf{P}\mathbf{A}) \vee \mathsf{P}\mathbf{B})) = \mathtt{T}$.
- So, we have $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof:

To show the second part, suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof:

To show the second part, suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.

- By property $\mathfrak{q}$, there is some $\mathsf{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ we have $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{b}) = \mathtt{T}$ iff $\mathsf{a} = \mathsf{b}$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof: To show the second part, suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.

- By property $\mathfrak{q}$, there is some $\mathsf{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ we have $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{b}) = \mathtt{T}$ iff $\mathsf{a} = \mathsf{b}$.
- Let $\mathsf{r} = \mathsf{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof: To show the second part, suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.

- By property $\mathfrak{q}$, there is some $\mathsf{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ we have $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{b}) = \mathtt{T}$ iff $\mathsf{a} = \mathsf{b}$.
- Let $\mathsf{r} = \mathsf{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})$.
- From $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$ we get $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\neg\mathsf{P}\mathbf{A} \lor \mathsf{P}\mathbf{B})) = \mathtt{T}$ (where $\mathsf{P}_{\alpha\to o} \notin \mathbf{Free}(\mathbf{A}) \cup \mathbf{Free}(\mathbf{B})$).

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof: To show the second part, suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.

- By property $\mathfrak{q}$, there is some $\mathsf{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ we have $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{b}) = \mathtt{T}$ iff $\mathsf{a} = \mathsf{b}$.
- Let $\mathsf{r} = \mathsf{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})$.
- From $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$ we get $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\neg\mathsf{P}\mathbf{A} \lor \mathsf{P}\mathbf{B})) = \mathtt{T}$ (where $\mathsf{P}_{\alpha\to o} \notin \mathbf{Free}(\mathbf{A}) \cup \mathbf{Free}(\mathbf{B})$).
- Since $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\mathsf{P}\mathbf{A})) = \upsilon(\mathsf{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})@\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$, we must have $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\mathsf{P}\mathbf{B})) = \mathtt{T}$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof: To show the second part, suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.

- By property $\mathfrak{q}$, there is some $\mathsf{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ we have $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{b}) = \mathtt{T}$ iff $\mathsf{a} = \mathsf{b}$.
- Let $\mathsf{r} = \mathsf{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})$.
- From $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$ we get $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\neg\mathsf{P}\mathbf{A} \lor \mathsf{P}\mathbf{B})) = \mathtt{T}$ (where $\mathsf{P}_{\alpha\to o} \notin \mathbf{Free}(\mathbf{A}) \cup \mathbf{Free}(\mathbf{B})$).
- Since $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\mathsf{P}\mathbf{A})) = \upsilon(\mathsf{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})@\mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$, we must have $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\mathsf{P}\mathbf{B})) = \mathtt{T}$.
- That is, $\upsilon(\mathsf{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})@\mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$, hence $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Extensionality and Leibniz Equality

Lemma: (Leibniz Equality in Σ-models) Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, φ be an assignment, $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$.

1. If $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.
2. If $\mathcal{M}$ satisf. $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

Proof:

To show the second part, suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$.

- By property $\mathfrak{q}$, there is some $\mathsf{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ we have $\upsilon(\mathsf{q}^\alpha @ \mathsf{a} @ \mathsf{b}) = \mathtt{T}$ iff $\mathsf{a} = \mathsf{b}$.
- Let $\mathsf{r} = \mathsf{q}^\alpha @ \mathcal{E}_\varphi(\mathbf{A})$.
- From $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) = \mathtt{T}$ we get $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\neg \mathsf{P}\mathbf{A} \lor \mathsf{P}\mathbf{B})) = \mathtt{T}$ (where $\mathsf{P}_{\alpha\to o} \notin \mathbf{Free}(\mathbf{A}) \cup \mathbf{Free}(\mathbf{B})$).
- Since $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\mathsf{P}\mathbf{A})) = \upsilon(\mathsf{q}^\alpha @ \mathcal{E}_\varphi(\mathbf{A}) @ \mathcal{E}_\varphi(\mathbf{A})) = \mathtt{T}$, we must have $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/\mathsf{P}]}(\mathsf{P}\mathbf{B})) = \mathtt{T}$.
- That is, $\upsilon(\mathsf{q}^\alpha @ \mathcal{E}_\varphi(\mathbf{A}) @ \mathcal{E}_\varphi(\mathbf{B})) = \mathtt{T}$, hence $\mathcal{E}_\varphi(\mathbf{A}) = \mathcal{E}_\varphi(\mathbf{B})$.

q.e.d.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathtt{EXT}^{\to}_{\doteq}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathtt{EXT}^{\to}_{\doteq}$.
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathtt{EXT}^{o}_{\doteq}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\to}_{\doteq}$.
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathrm{EXT}^{o}_{\doteq}$.
3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathrm{EXT}^{\to}_{\doteq}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\to}_{\doteq}$.
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathrm{EXT}^{o}_{\doteq}$.
3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathrm{EXT}^{\to}_{\doteq}$.
4. If $\mathcal{M}$ satisfies $\mathfrak{b}$, then $\mathcal{M} \models \mathrm{EXT}^{o}_{\doteq}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\to}_{\doteq}$.
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathrm{EXT}^{o}_{\doteq}$.
3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathrm{EXT}^{\to}_{\doteq}$.
4. If $\mathcal{M}$ satisfies $\mathfrak{b}$, then $\mathcal{M} \models \mathrm{EXT}^{o}_{\doteq}$.

in	$\mathfrak{M}_{\beta}(\Sigma)$, $\mathfrak{M}_{\beta\eta}(\Sigma)$, $\mathfrak{M}_{\beta\xi}(\Sigma)$		$\mathfrak{M}_{\beta\mathfrak{f}}(\Sigma)$		$\mathfrak{M}_{\beta\mathfrak{b}}(\Sigma)$,$\mathfrak{M}_{\beta\eta\mathfrak{b}}(\Sigma)$,$\mathfrak{M}_{\beta\xi\mathfrak{b}}(\Sigma)$		$\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}(\Sigma)$	
formula	valid?	by	valid?	by	valid?	by	valid?	by
$\mathrm{EXT}^{\to}_{\doteq}$	—	1.	+	3.	—	1.	+	3.
$\mathrm{EXT}^{o}_{\doteq}$	—	2.	—	2.	+	4.	+	4.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\to}_{\doteq}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{f}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\rightarrow}_{\doteq}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{f}$.

- Then there must be types α and β and objects $f, g \in \mathcal{D}_{\alpha\to\beta}$ such that $f \neq g$ but $f@a = g@a$ for every $a \in \mathcal{D}_\alpha$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\rightarrow}_{\doteq}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{f}$.

- Then there must be types α and β and objects $f, g \in \mathcal{D}_{\alpha\to\beta}$ such that $f \neq g$ but $f@a = g@a$ for every $a \in \mathcal{D}_\alpha$.
- Let $F_{\alpha\to\beta}, G_{\alpha\to\beta} \in \mathcal{V}_{\alpha\to\beta}$ be distinct variables, $X_\alpha \in \mathcal{V}_\alpha$, and φ be any assignment with $\varphi(F) = f$ and $\varphi(G) = g$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\rightarrow}_{\doteq}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{f}$.

- Then there must be types α and β and objects $f, g \in \mathcal{D}_{\alpha\to\beta}$ such that $f \neq g$ but $f@a = g@a$ for every $a \in \mathcal{D}_\alpha$.
- Let $F_{\alpha\to\beta}, G_{\alpha\to\beta} \in \mathcal{V}_{\alpha\to\beta}$ be distinct variables, $X_\alpha \in \mathcal{V}_\alpha$, and φ be any assignment with $\varphi(F) = f$ and $\varphi(G) = g$.
- For any $a \in \mathcal{D}_\alpha$, $f@a = g@a$ implies $\mathcal{E}_{\varphi,[a/X]}(FX) = \mathcal{E}_{\varphi,[a/X]}(GX)$ implies $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq^{\beta} GX)) = \mathtt{T}$ by Lemma 'Leibniz Equality in Σ-models(1.)'.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\rightarrow}_{\doteq}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{f}$.

- Then there must be types α and β and objects $f, g \in \mathcal{D}_{\alpha\to\beta}$ such that $f \neq g$ but $f@a = g@a$ for every $a \in \mathcal{D}_\alpha$.
- Let $F_{\alpha\to\beta}, G_{\alpha\to\beta} \in \mathcal{V}_{\alpha\to\beta}$ be distinct variables, $X_\alpha \in \mathcal{V}_\alpha$, and φ be any assignment with $\varphi(F) = f$ and $\varphi(G) = g$.
- For any $a \in \mathcal{D}_\alpha$, $f@a = g@a$ implies $\mathcal{E}_{\varphi,[a/X]}(FX) = \mathcal{E}_{\varphi,[a/X]}(GX)$ implies $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq^{\beta} GX)) = \mathtt{T}$ by Lemma 'Leibniz Equality in Σ-models(1.)'.
- Hence, we have $\upsilon(\mathcal{E}_{\varphi}(\forall X.(FX \doteq^{\beta} GX))) = \mathtt{T}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathtt{EXT}^{\rightarrow}_{\doteq}$.

Proof:

Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{f}$.

- Then there must be types α and β and objects $f, g \in \mathcal{D}_{\alpha\to\beta}$ such that $f \neq g$ but $f@a = g@a$ for every $a \in \mathcal{D}_\alpha$.
- Let $F_{\alpha\to\beta}, G_{\alpha\to\beta} \in \mathcal{V}_{\alpha\to\beta}$ be distinct variables, $X_\alpha \in \mathcal{V}_\alpha$, and φ be any assignment with $\varphi(F) = f$ and $\varphi(G) = g$.
- For any $a \in \mathcal{D}_\alpha$, $f@a = g@a$ implies $\mathcal{E}_{\varphi,[a/X]}(FX) = \mathcal{E}_{\varphi,[a/X]}(GX)$ implies $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq^{\beta} GX)) = \mathtt{T}$ by Lemma 'Leibniz Equality in Σ-models(1.)'.
- Hence, we have $\upsilon(\mathcal{E}_\varphi(\forall X.(FX \doteq^{\beta} GX))) = \mathtt{T}$.
- On the other hand, since $f \neq g$ and $\mathcal{M}$ satisfies property $\mathfrak{q}$, we have $\upsilon(\mathcal{E}_\varphi(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ by contraposition of Lemma 'Leibniz Equality in Σ-models(2.)'.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathtt{EXT}^{\rightarrow}_{\doteq}$.

Proof:

Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{f}$.

- Then there must be types α and β and objects $f, g \in \mathcal{D}_{\alpha\to\beta}$ such that $f \neq g$ but $f@a = g@a$ for every $a \in \mathcal{D}_\alpha$.
- Let $F_{\alpha\to\beta}, G_{\alpha\to\beta} \in \mathcal{V}_{\alpha\to\beta}$ be distinct variables, $X_\alpha \in \mathcal{V}_\alpha$, and φ be any assignment with $\varphi(F) = f$ and $\varphi(G) = g$.
- For any $a \in \mathcal{D}_\alpha$, $f@a = g@a$ implies $\mathcal{E}_{\varphi,[a/X]}(FX) = \mathcal{E}_{\varphi,[a/X]}(GX)$ implies $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq^{\beta} GX)) = \mathtt{T}$ by Lemma 'Leibniz Equality in Σ-models(1.)'.
- Hence, we have $\upsilon(\mathcal{E}_\varphi(\forall X.(FX \doteq^{\beta} GX))) = \mathtt{T}$.
- On the other hand, since $f \neq g$ and $\mathcal{M}$ satisfies property $\mathfrak{q}$, we have $\upsilon(\mathcal{E}_\varphi(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ by contraposition of Lemma 'Leibniz Equality in Σ-models(2.)'.
- This implies $\mathcal{M} \not\models \mathtt{EXT}^{\alpha\to\beta}_{\doteq}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathtt{EXT}^{o}_{\doteq}$.

Proof:

Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{b}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)

Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathtt{EXT}^{o}_{\doteq}$.

Proof:

Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{b}$.

- Then, there must be at least three elements in $\mathcal{D}_o$. Since υ maps into a two element set, there must be two distinct elements $a, b \in \mathcal{D}_o$ such that $\upsilon(a) = \upsilon(b)$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathtt{EXT}^{o}_{\doteq}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{b}$.

- Then, there must be at least three elements in $\mathcal{D}_o$. Since υ maps into a two element set, there must be two distinct elements $a, b \in \mathcal{D}_o$ such that $\upsilon(a) = \upsilon(b)$.
- Let $A_o, B_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$ with $\varphi(A) = a$ and $\varphi(B) = b$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathtt{EXT}^{o}_{\doteq}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{b}$.

- Then, there must be at least three elements in $\mathcal{D}_o$. Since υ maps into a two element set, there must be two distinct elements $a, b \in \mathcal{D}_o$ such that $\upsilon(a) = \upsilon(b)$.
- Let $A_o, B_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$ with $\varphi(A) = a$ and $\varphi(B) = b$.
- By Lemma 'Equivalence', we know $\upsilon(\mathcal{E}_\varphi(A \Leftrightarrow B)) = \mathtt{T}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathtt{EXT}^{o}_{\doteq}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{b}$.

- Then, there must be at least three elements in $\mathcal{D}_o$. Since υ maps into a two element set, there must be two distinct elements $a, b \in \mathcal{D}_o$ such that $\upsilon(a) = \upsilon(b)$.
- Let $A_o, B_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$ with $\varphi(A) = a$ and $\varphi(B) = b$.
- By Lemma 'Equivalence', we know $\upsilon(\mathcal{E}_\varphi(A \Leftrightarrow B)) = \mathtt{T}$.
- Since $a \neq b$ and property $\mathfrak{q}$ holds, by contraposition of Lemma 'Leibniz Equality in Σ-models(2.)', we know $\upsilon(\mathcal{E}_\varphi(A \doteq^{o} B)) = \mathtt{F}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathtt{EXT}^{o}_{\doteq}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{b}$.

- Then, there must be at least three elements in $\mathcal{D}_o$. Since υ maps into a two element set, there must be two distinct elements $a, b \in \mathcal{D}_o$ such that $\upsilon(a) = \upsilon(b)$.
- Let $A_o, B_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$ with $\varphi(A) = a$ and $\varphi(B) = b$.
- By Lemma 'Equivalence', we know $\upsilon(\mathcal{E}_\varphi(A \Leftrightarrow B)) = \mathtt{T}$.
- Since $a \neq b$ and property $\mathfrak{q}$ holds, by contraposition of Lemma 'Leibniz Equality in Σ-models(2.)', we know $\upsilon(\mathcal{E}_\varphi(A \doteq^{o} B)) = \mathtt{F}$.
- It follows that $\mathcal{M} \not\models \mathtt{EXT}^{o}_{\doteq}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathsf{EXT}^{o}_{\dot{=}}$.

Proof: Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{b}$.

- Then, there must be at least three elements in $\mathcal{D}_o$. Since υ maps into a two element set, there must be two distinct elements $a, b \in \mathcal{D}_o$ such that $\upsilon(a) = \upsilon(b)$.
- Let $A_o, B_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$ with $\varphi(A) = a$ and $\varphi(B) = b$.
- By Lemma 'Equivalence', we know $\upsilon(\mathcal{E}_\varphi(A \Leftrightarrow B)) = \mathtt{T}$.
- Since $a \neq b$ and property $\mathfrak{q}$ holds, by contraposition of Lemma 'Leibniz Equality in Σ-models(2.)', we know $\upsilon(\mathcal{E}_\varphi(A \doteq^{o} B)) = \mathtt{F}$.
- It follows that $\mathcal{M} \not\models \mathsf{EXT}^{o}_{\dot{=}}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathsf{EXT}^{\to}_{\dot{=}}$.

Proof: Let φ be any assignment into $\mathcal{M}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathsf{EXT}^{\to}_{\dot{=}}$.

Proof: Let φ be any assignment into $\mathcal{M}$.

- From $\upsilon(\mathcal{E}_\varphi(\forall X_\alpha . FX \doteq GX)) = \mathtt{T}$ we know $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq GX)) = \mathtt{T}$ holds for all $a \in \mathcal{D}_\alpha$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathsf{EXT}^{\to}_{\dot{=}}$.

Proof: Let φ be any assignment into $\mathcal{M}$.

- From $\upsilon(\mathcal{E}_\varphi(\forall X_\alpha . FX \doteq GX)) = \mathtt{T}$ we know $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq GX)) = \mathtt{T}$ holds for all $a \in \mathcal{D}_\alpha$.
- By Lemma 'Leibniz Equality in Σ-models(2.)' we can conclude that $\mathcal{E}_{\varphi,[a/X]}(FX) = \mathcal{E}_{\varphi,[a/X]}(GX)$ for all $a \in \mathcal{D}_\alpha$ and hence $\mathcal{E}_{\varphi,[a/X]}(F)@\mathcal{E}_{\varphi,[a/X]}(X) = \mathcal{E}_{\varphi,[a/X]}(G)@\mathcal{E}_{\varphi,[a/X]}(X)$ for all $a \in \mathcal{D}_\alpha$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathrm{EXT}^{\to}_{\dot=}$.

Proof:
Let φ be any assignment into $\mathcal{M}$.

- From $\upsilon(\mathcal{E}_\varphi(\forall X_\alpha . FX \doteq GX)) = \mathtt{T}$ we know $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq GX)) = \mathtt{T}$ holds for all $a \in \mathcal{D}_\alpha$.
- By Lemma 'Leibniz Equality in Σ-models(2.)' we can conclude that $\mathcal{E}_{\varphi,[a/X]}(FX) = \mathcal{E}_{\varphi,[a/X]}(GX)$ for all $a \in \mathcal{D}_\alpha$ and hence $\mathcal{E}_{\varphi,[a/X]}(F)@\mathcal{E}_{\varphi,[a/X]}(X) = \mathcal{E}_{\varphi,[a/X]}(G)@\mathcal{E}_{\varphi,[a/X]}(X)$ for all $a \in \mathcal{D}_\alpha$.
- That is, $\mathcal{E}_{\varphi,[a/X]}(F)@a = \mathcal{E}_{\varphi,[a/X]}(G)@a$ for all $a \in \mathcal{D}_\alpha$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathrm{EXT}^{\to}_{\dot=}$.

Proof:
Let φ be any assignment into $\mathcal{M}$.

- From $\upsilon(\mathcal{E}_\varphi(\forall X_\alpha . FX \doteq GX)) = \mathtt{T}$ we know $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq GX)) = \mathtt{T}$ holds for all $a \in \mathcal{D}_\alpha$.
- By Lemma 'Leibniz Equality in Σ-models(2.)' we can conclude that $\mathcal{E}_{\varphi,[a/X]}(FX) = \mathcal{E}_{\varphi,[a/X]}(GX)$ for all $a \in \mathcal{D}_\alpha$ and hence $\mathcal{E}_{\varphi,[a/X]}(F)@\mathcal{E}_{\varphi,[a/X]}(X) = \mathcal{E}_{\varphi,[a/X]}(G)@\mathcal{E}_{\varphi,[a/X]}(X)$ for all $a \in \mathcal{D}_\alpha$.
- That is, $\mathcal{E}_{\varphi,[a/X]}(F)@a = \mathcal{E}_{\varphi,[a/X]}(G)@a$ for all $a \in \mathcal{D}_\alpha$.
- Since X does not occur free in F or G, by property $\mathfrak{f}$ and Definition of Σ-evaluations we obtain $\mathcal{E}_\varphi(F) = \mathcal{E}_\varphi(G)$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathrm{EXT}^{\to}_{\dot=}$.

Proof:
Let φ be any assignment into $\mathcal{M}$.

- From $\upsilon(\mathcal{E}_\varphi(\forall X_\alpha . FX \doteq GX)) = \mathtt{T}$ we know $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq GX)) = \mathtt{T}$ holds for all $a \in \mathcal{D}_\alpha$.
- By Lemma 'Leibniz Equality in Σ-models(2.)' we can conclude that $\mathcal{E}_{\varphi,[a/X]}(FX) = \mathcal{E}_{\varphi,[a/X]}(GX)$ for all $a \in \mathcal{D}_\alpha$ and hence $\mathcal{E}_{\varphi,[a/X]}(F)@\mathcal{E}_{\varphi,[a/X]}(X) = \mathcal{E}_{\varphi,[a/X]}(G)@\mathcal{E}_{\varphi,[a/X]}(X)$ for all $a \in \mathcal{D}_\alpha$.
- That is, $\mathcal{E}_{\varphi,[a/X]}(F)@a = \mathcal{E}_{\varphi,[a/X]}(G)@a$ for all $a \in \mathcal{D}_\alpha$.
- Since X does not occur free in F or G, by property $\mathfrak{f}$ and Definition of Σ-evaluations we obtain $\mathcal{E}_\varphi(F) = \mathcal{E}_\varphi(G)$.
- This finally gives us that $\upsilon(\mathcal{E}_\varphi(F \doteq^{\alpha\to\beta} G)) = \mathtt{T}$ with Lemma 'Leibniz Equality in Σ-models(1.)'.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathrm{EXT}^{\to}_{\dot=}$.

Proof:
Let φ be any assignment into $\mathcal{M}$.

- From $\upsilon(\mathcal{E}_\varphi(\forall X_\alpha . FX \doteq GX)) = \mathtt{T}$ we know $\upsilon(\mathcal{E}_{\varphi,[a/X]}(FX \doteq GX)) = \mathtt{T}$ holds for all $a \in \mathcal{D}_\alpha$.
- By Lemma 'Leibniz Equality in Σ-models(2.)' we can conclude that $\mathcal{E}_{\varphi,[a/X]}(FX) = \mathcal{E}_{\varphi,[a/X]}(GX)$ for all $a \in \mathcal{D}_\alpha$ and hence $\mathcal{E}_{\varphi,[a/X]}(F)@\mathcal{E}_{\varphi,[a/X]}(X) = \mathcal{E}_{\varphi,[a/X]}(G)@\mathcal{E}_{\varphi,[a/X]}(X)$ for all $a \in \mathcal{D}_\alpha$.
- That is, $\mathcal{E}_{\varphi,[a/X]}(F)@a = \mathcal{E}_{\varphi,[a/X]}(G)@a$ for all $a \in \mathcal{D}_\alpha$.
- Since X does not occur free in F or G, by property $\mathfrak{f}$ and Definition of Σ-evaluations we obtain $\mathcal{E}_\varphi(F) = \mathcal{E}_\varphi(G)$.
- This finally gives us that $\upsilon(\mathcal{E}_\varphi(F \doteq^{\alpha\to\beta} G)) = \mathtt{T}$ with Lemma 'Leibniz Equality in Σ-models(1.)'.
- It follows that $\mathcal{M} \models \mathrm{EXT}^{\alpha\to\beta}_{\dot=}$ and $\mathcal{M} \models \mathrm{EXT}^{\to}_{\dot=}$, since α and β were chosen arbitrarily.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

4. If $\mathcal{M}$ satisfies $\mathfrak{b}$, then $\mathcal{M} \models \mathsf{EXT}^{o}_{\underline{\dot{=}}}$.

Proof: Let $\mathsf{A}_o, \mathsf{B}_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

4. If $\mathcal{M}$ satisfies $\mathfrak{b}$, then $\mathcal{M} \models \mathsf{EXT}^{o}_{\underline{\dot{=}}}$.

Proof: Let $\mathsf{A}_o, \mathsf{B}_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$.

- Since property $\mathfrak{b}$ holds, we can assume $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ and υ is the identity function.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

4. If $\mathcal{M}$ satisfies $\mathfrak{b}$, then $\mathcal{M} \models \mathsf{EXT}^{o}_{\underline{\dot{=}}}$.

Proof: Let $\mathsf{A}_o, \mathsf{B}_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$.

- Since property $\mathfrak{b}$ holds, we can assume $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ and υ is the identity function.
- Suppose $\upsilon(\mathcal{E}_\varphi(\mathsf{A} \Leftrightarrow \mathsf{B})) = \mathtt{T}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

4. If $\mathcal{M}$ satisfies $\mathfrak{b}$, then $\mathcal{M} \models \mathsf{EXT}^{o}_{\underline{\dot{=}}}$.

Proof: Let $\mathsf{A}_o, \mathsf{B}_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$.

- Since property $\mathfrak{b}$ holds, we can assume $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ and υ is the identity function.
- Suppose $\upsilon(\mathcal{E}_\varphi(\mathsf{A} \Leftrightarrow \mathsf{B})) = \mathtt{T}$.
- By Lemma 'Equivalence', we have $\mathcal{E}_\varphi(\mathsf{A}) = \upsilon(\mathcal{E}_\varphi(\mathsf{A})) = \upsilon(\mathcal{E}_\varphi(\mathsf{B})) = \mathcal{E}_\varphi(\mathsf{B})$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

4. If $\mathcal{M}$ satisfies $\mathfrak{b}$, then $\mathcal{M} \models \mathrm{EXT}^{o}_{\doteq}$.

Proof: Let $A_o, B_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$.

- Since property $\mathfrak{b}$ holds, we can assume $\mathcal{D}_o = \{\mathtt{T}, \mathtt{F}\}$ and υ is the identity function.
- Suppose $\upsilon(\mathcal{E}_\varphi(A \Leftrightarrow B)) = \mathtt{T}$.
- By Lemma 'Equivalence', we have $\mathcal{E}_\varphi(A) = \upsilon(\mathcal{E}_\varphi(A)) = \upsilon(\mathcal{E}_\varphi(B)) = \mathcal{E}_\varphi(B)$.
- By Lemma 'Leibniz Equality in Σ-models(1.)', we have $\upsilon(\mathcal{E}_\varphi(A \doteq^{o} B)) = \mathtt{T}$.

Extensionality and Leibniz Equality

Thm.: (Extensionality in Σ-models)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\rightarrow}_{\doteq}$.
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathrm{EXT}^{o}_{\doteq}$.
3. If $\mathcal{M}$ satisfies $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathrm{EXT}^{\rightarrow}_{\doteq}$.
4. If $\mathcal{M}$ satisfies $\mathfrak{b}$, then $\mathcal{M} \models \mathrm{EXT}^{o}_{\doteq}$.

Proof:

q.e.d.

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o . \forall B_o . A \doteq^{o} B \Rightarrow (A \Leftrightarrow B)$.

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o \centerdot \forall B_o \centerdot A \doteq^o B \Rightarrow (A \Leftrightarrow B)$.
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall F_{\alpha\to\beta} \centerdot \forall G_{\alpha\to\beta} \centerdot F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha \centerdot FX \doteq^\beta GX)$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o \centerdot \forall B_o \centerdot A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall F_{\alpha\to\beta} \centerdot \forall G_{\alpha\to\beta} \centerdot F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha \centerdot FX \doteq^\beta GX)$

Proof: (1.)

$\upsilon(\mathcal{E}_\varphi(\forall A_o \centerdot \forall B_o \centerdot A \doteq^o B \Rightarrow (A \Leftrightarrow B))) = \mathtt{T}$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o \centerdot \forall B_o \centerdot A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall F_{\alpha\to\beta} \centerdot \forall G_{\alpha\to\beta} \centerdot F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha \centerdot FX \doteq^\beta GX)$

Proof: (1.)

$\upsilon(\mathcal{E}_\varphi(\forall A_o \centerdot \forall B_o \centerdot A \doteq^o B \Rightarrow (A \Leftrightarrow B))) = \mathtt{T}$

- iff (for all $a, b \in \mathcal{D}_\alpha$) $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \doteq^o B) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \Leftrightarrow B)) = \mathtt{T}$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o \centerdot \forall B_o \centerdot A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall F_{\alpha\to\beta} \centerdot \forall G_{\alpha\to\beta} \centerdot F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha \centerdot FX \doteq^\beta GX)$

Proof: (1.)

$\upsilon(\mathcal{E}_\varphi(\forall A_o \centerdot \forall B_o \centerdot A \doteq^o B \Rightarrow (A \Leftrightarrow B))) = \mathtt{T}$

- iff (for all $a, b \in \mathcal{D}_\alpha$) $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \doteq^o B) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \Leftrightarrow B)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \doteq^o B)) = \mathtt{T}$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then
   $\mathcal{M} \models \forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX)$

Proof: (1.)

$\upsilon(\mathcal{E}_\varphi(\forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B))) = \mathtt{T}$

- iff (for all $a, b \in \mathcal{D}_\alpha$)
  $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \doteq^o B)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \Leftrightarrow B)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \doteq^o B)) = \mathtt{T}$
- then by Lemma 'Leibniz Equality in Σ-models(2.)':
  $\mathcal{E}_{\varphi[a/A][b/B]}(A) = \mathcal{E}_{\varphi[a/A][b/B]}(B)$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then
   $\mathcal{M} \models \forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX)$

Proof: (1.)

$\upsilon(\mathcal{E}_\varphi(\forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B))) = \mathtt{T}$

- iff (for all $a, b \in \mathcal{D}_\alpha$)
  $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \doteq^o B)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \Leftrightarrow B)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \doteq^o B)) = \mathtt{T}$
- then by Lemma 'Leibniz Equality in Σ-models(2.)':
  $\mathcal{E}_{\varphi[a/A][b/B]}(A) = \mathcal{E}_{\varphi[a/A][b/B]}(B)$
- then by Lemma 'Equivalence': $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \Leftrightarrow B)) = \mathtt{T}$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then
   $\mathcal{M} \models \forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX)$

Proof: (1.)

$\upsilon(\mathcal{E}_\varphi(\forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B))) = \mathtt{T}$

- iff (for all $a, b \in \mathcal{D}_\alpha$)
  $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \doteq^o B)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \Leftrightarrow B)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \doteq^o B)) = \mathtt{T}$
- then by Lemma 'Leibniz Equality in Σ-models(2.)':
  $\mathcal{E}_{\varphi[a/A][b/B]}(A) = \mathcal{E}_{\varphi[a/A][b/B]}(B)$
- then by Lemma 'Equivalence': $\upsilon(\mathcal{E}_{\varphi[a/A][b/B]}(A \Leftrightarrow B)) = \mathtt{T}$

q.e.d.

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then
   $\mathcal{M} \models \forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX)$

Proof: (2.) $\upsilon(\mathcal{E}_\varphi(\forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX))) = \mathtt{T}$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then
   $\mathcal{M} \models \forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX)$

Proof: (2.) $\upsilon(\mathcal{E}_\varphi(\forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX))) = \mathtt{T}$

- iff (for all $f, g \in \mathcal{D}_{\alpha\to\beta}$)
  $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(\forall X_\alpha. FX \doteq^\beta GX)) = \mathtt{T}$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then
   $\mathcal{M} \models \forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX)$

Proof: (2.) $\upsilon(\mathcal{E}_\varphi(\forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX))) = \mathtt{T}$

- iff (for all $f, g \in \mathcal{D}_{\alpha\to\beta}$)
  $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(\forall X_\alpha. FX \doteq^\beta GX)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{T}$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then
   $\mathcal{M} \models \forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX)$

Proof: (2.) $\upsilon(\mathcal{E}_\varphi(\forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX))) = \mathtt{T}$

- iff (for all $f, g \in \mathcal{D}_{\alpha\to\beta}$)
  $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(\forall X_\alpha. FX \doteq^\beta GX)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{T}$
- by Lemma 'Leibniz Equality in Σ-models(2.)': $\mathcal{E}_{\varphi[f/F][g/G]}(F) = \mathcal{E}_{\varphi[f/F][g/G]}(G)$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o. \forall B_o. A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then
   $\mathcal{M} \models \forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX)$

Proof: (2.) $\upsilon(\mathcal{E}_\varphi(\forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha. FX \doteq^\beta GX))) = \mathtt{T}$

- iff (for all $f, g \in \mathcal{D}_{\alpha\to\beta}$)
  $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(\forall X_\alpha. FX \doteq^\beta GX)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{T}$
- by Lemma 'Leibniz Equality in Σ-models(2.)': $\mathcal{E}_{\varphi[f/F][g/G]}(F) = \mathcal{E}_{\varphi[f/F][g/G]}(G)$
- X not free in G or F (for any $a \in \mathcal{D}_\alpha$): $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(F) = \mathcal{E}_{\varphi[f/F][g/G][a/X]}(G)$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o . \forall B_o . A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall F_{\alpha\to\beta} . \forall G_{\alpha\to\beta} . F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha . FX \doteq^\beta GX)$

Proof: (2.) $\upsilon(\mathcal{E}_\varphi(\forall F_{\alpha\to\beta} . \forall G_{\alpha\to\beta} . F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha . FX \doteq^\beta GX))) = \mathtt{T}$

- iff (for all $f, g \in \mathcal{D}_{\alpha\to\beta}$) $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(\forall X_\alpha . FX \doteq^\beta GX)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{T}$
- by Lemma 'Leibniz Equality in Σ-models(2.)': $\mathcal{E}_{\varphi[f/F][g/G]}(F) = \mathcal{E}_{\varphi[f/F][g/G]}(G)$
- X not free in G or F (for any $a \in \mathcal{D}_\alpha$): $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(F) = \mathcal{E}_{\varphi[f/F][g/G][a/X]}(G)$
- furthermore $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(F) @ \mathcal{E}_{\varphi[f/F][g/G][a/X]}(X) = \mathcal{E}_{\varphi[f/F][g/G]}(G) @ \mathcal{E}_{\varphi[f/F][g/G][a/X]}(X)$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o . \forall B_o . A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall F_{\alpha\to\beta} . \forall G_{\alpha\to\beta} . F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha . FX \doteq^\beta GX)$

Proof: (2.) $\upsilon(\mathcal{E}_\varphi(\forall F_{\alpha\to\beta} . \forall G_{\alpha\to\beta} . F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha . FX \doteq^\beta GX))) = \mathtt{T}$

- iff (for all $f, g \in \mathcal{D}_{\alpha\to\beta}$) $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(\forall X_\alpha . FX \doteq^\beta GX)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{T}$
- by Lemma 'Leibniz Equality in Σ-models(2.)': $\mathcal{E}_{\varphi[f/F][g/G]}(F) = \mathcal{E}_{\varphi[f/F][g/G]}(G)$
- X not free in G or F (for any $a \in \mathcal{D}_\alpha$): $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(F) = \mathcal{E}_{\varphi[f/F][g/G][a/X]}(G)$
- furthermore $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(F) @ \mathcal{E}_{\varphi[f/F][g/G][a/X]}(X) = \mathcal{E}_{\varphi[f/F][g/G]}(G) @ \mathcal{E}_{\varphi[f/F][g/G][a/X]}(X)$
- thus $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(FX) = \mathcal{E}_{\varphi[f/F][g/G][a/X]}(GX)$

Extensionality and Leibniz Equality

Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o . \forall B_o . A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall F_{\alpha\to\beta} . \forall G_{\alpha\to\beta} . F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha . FX \doteq^\beta GX)$

Proof: (2.) $\upsilon(\mathcal{E}_\varphi(\forall F_{\alpha\to\beta} . \forall G_{\alpha\to\beta} . F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha . FX \doteq^\beta GX))) = \mathtt{T}$

- iff (for all $f, g \in \mathcal{D}_{\alpha\to\beta}$) $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(\forall X_\alpha . FX \doteq^\beta GX)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{T}$
- by Lemma 'Leibniz Equality in Σ-models(2.)': $\mathcal{E}_{\varphi[f/F][g/G]}(F) = \mathcal{E}_{\varphi[f/F][g/G]}(G)$
- X not free in G or F (for any $a \in \mathcal{D}_\alpha$): $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(F) = \mathcal{E}_{\varphi[f/F][g/G][a/X]}(G)$
- furthermore $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(F) @ \mathcal{E}_{\varphi[f/F][g/G][a/X]}(X) = \mathcal{E}_{\varphi[f/F][g/G]}(G) @ \mathcal{E}_{\varphi[f/F][g/G][a/X]}(X)$
- thus $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(FX) = \mathcal{E}_{\varphi[f/F][g/G][a/X]}(GX)$
- thus $\upsilon(\mathcal{E}_{\varphi[f/F][g/G][a/X]}(FX \doteq^\beta GX)) = \mathtt{T}$ by Lemma ...

Extensionality and Leibniz Equality

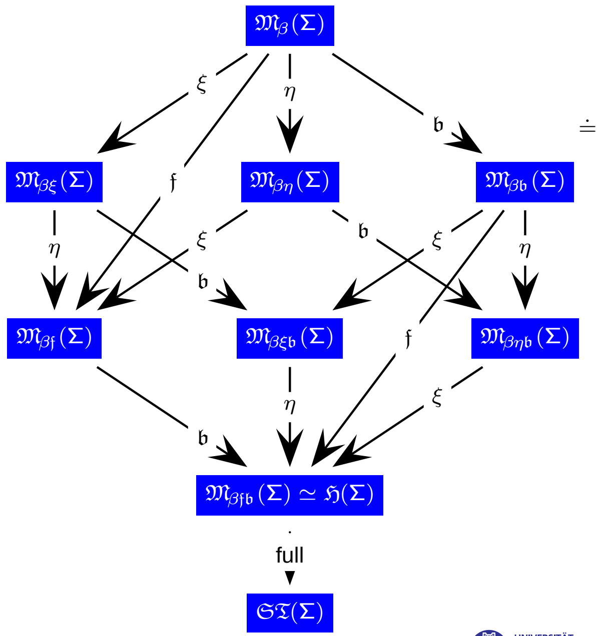
Thm.: (Trivial Extensionality Directions in Σ-Models)

1. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall A_o . \forall B_o . A \doteq^o B \Rightarrow (A \Leftrightarrow B)$
2. If $\mathcal{M}$ satisfies $\mathfrak{q}$, then $\mathcal{M} \models \forall F_{\alpha\to\beta} . \forall G_{\alpha\to\beta} . F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha . FX \doteq^\beta GX)$

Proof: (2.) $\upsilon(\mathcal{E}_\varphi(\forall F_{\alpha\to\beta} . \forall G_{\alpha\to\beta} . F \doteq^{\alpha\to\beta} G \Rightarrow (\forall X_\alpha . FX \doteq^\beta GX))) = \mathtt{T}$

- iff (for all $f, g \in \mathcal{D}_{\alpha\to\beta}$) $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{F}$ or $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(\forall X_\alpha . FX \doteq^\beta GX)) = \mathtt{T}$
- assume $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(F \doteq^{\alpha\to\beta} G)) = \mathtt{T}$
- by Lemma 'Leibniz Equality in Σ-models(2.)': $\mathcal{E}_{\varphi[f/F][g/G]}(F) = \mathcal{E}_{\varphi[f/F][g/G]}(G)$
- X not free in G or F (for any $a \in \mathcal{D}_\alpha$): $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(F) = \mathcal{E}_{\varphi[f/F][g/G][a/X]}(G)$
- furthermore $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(F) @ \mathcal{E}_{\varphi[f/F][g/G][a/X]}(X) = \mathcal{E}_{\varphi[f/F][g/G]}(G) @ \mathcal{E}_{\varphi[f/F][g/G][a/X]}(X)$
- thus $\mathcal{E}_{\varphi[f/F][g/G][a/X]}(FX) = \mathcal{E}_{\varphi[f/F][g/G][a/X]}(GX)$
- thus $\upsilon(\mathcal{E}_{\varphi[f/F][g/G][a/X]}(FX \doteq^\beta GX)) = \mathtt{T}$ by Lemma ...
- thus $\upsilon(\mathcal{E}_{\varphi[f/F][g/G]}(\forall X_\alpha . FX \doteq^\beta GX)) = \mathtt{T}$ q.e.d.

Leibniz Equality in Σ-Models

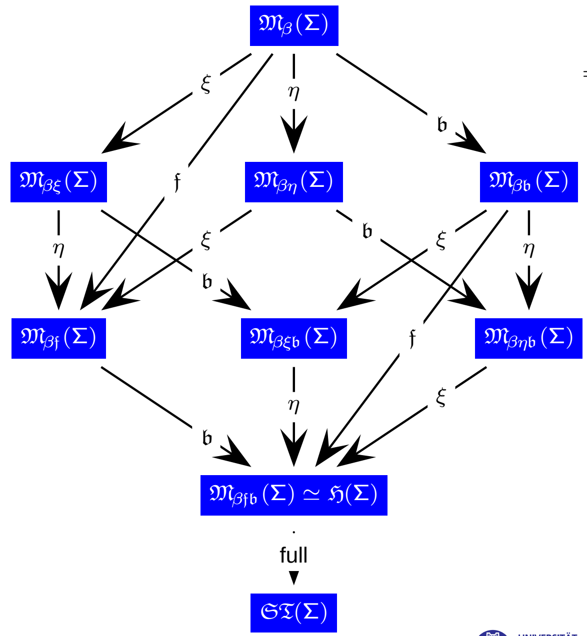


$\doteq$ **is equivalence relation**

- $\forall X_\alpha.X \doteq X$

Proof:
$\upsilon(\mathcal{E}_\varphi(\forall X_\alpha.X \doteq X)) = \mathtt{T}$
iff $\upsilon(\mathcal{E}_{\varphi[a/X]}(X \doteq X)) = \mathtt{T}$ **for all** $a \in \mathbf{D}_\alpha$
holds by Lemma 'Leibniz Equality in Σ**-models(1.)'**
since $\mathcal{E}_{\varphi[a/X]}(X) = \mathcal{E}_{\varphi[a/X]}(X)$ **for all** $a \in \mathbf{D}_\alpha$.

Leibniz Equality in Σ-Models

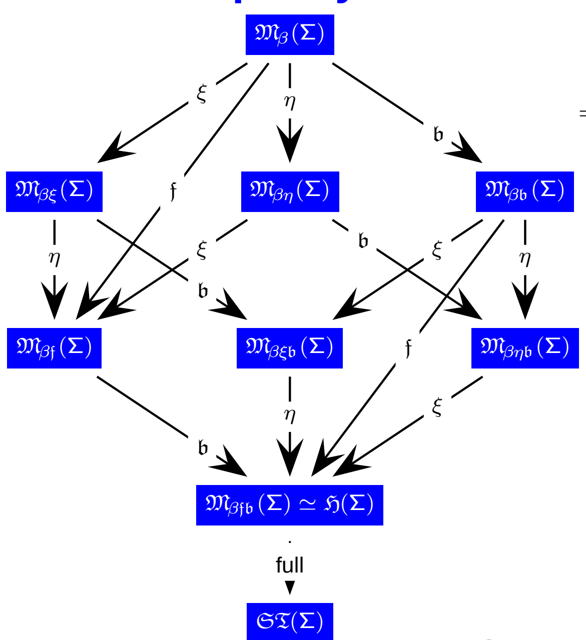


$\doteq$ **is equivalence relation**

- $\forall X_\alpha.X \doteq X$
- $\forall X_\alpha, Y_\alpha.X \doteq Y \Rightarrow Y \doteq X$

Proof:
$\upsilon(\mathcal{E}_\varphi(\forall X_\alpha, Y_\alpha.X \doteq Y \Leftrightarrow Y \doteq X)) = \mathtt{T}$
iff $\upsilon(\mathcal{E}_{\varphi[a/X][b/X]}(X \doteq Y)) = \mathtt{F}$ **or** $\upsilon(\mathcal{E}_{\varphi[a/X][b/X]}(Y \doteq X)) = \mathtt{T}$ **for all** $a, b \in \mathbf{D}_\alpha$
assume $\upsilon(\mathcal{E}_{\varphi[a/X][b/X]}(X \doteq Y)) = \mathtt{T}$
by Lemma 'Leibniz Equality in Σ**-models(2.)'**
$\mathcal{E}_{\varphi[a/X]}(X) = \mathcal{E}_{\varphi[a/X]}(Y)$ **for all** $a, b \in \mathbf{D}_\alpha$
hence $\mathcal{E}_{\varphi[a/X]}(Y) = \mathcal{E}_{\varphi[a/X]}(X)$ **for all** $a, b \in \mathbf{D}_\alpha$
hence $\upsilon(\mathcal{E}_{\varphi[a/X][b/X]}(X \doteq Y)) = \mathtt{T}$ **for all** $a, b \in \mathbf{D}_\alpha$ **by Lemma 'Leibniz Equality in** Σ**-models(2.)'**

Leibniz Equality in Σ-Models

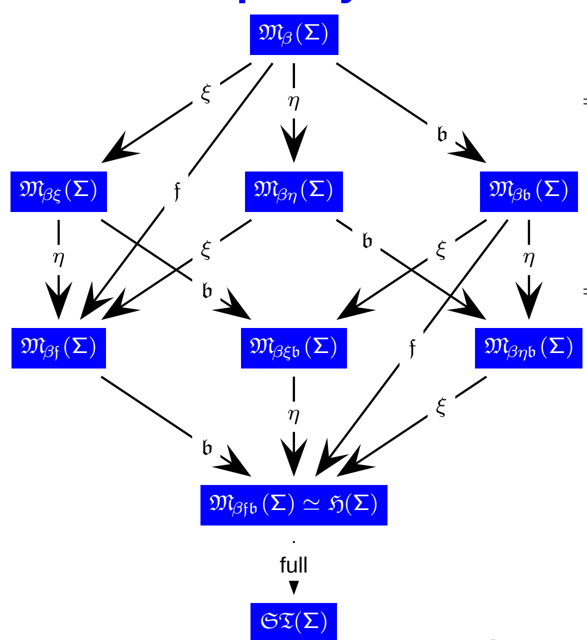


$\doteq$ **is equivalence relation**

- $\forall X_\alpha.X \doteq X$
- $\forall X_\alpha, Y_\alpha.X \doteq Y \Rightarrow Y \doteq X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha.(X \doteq Y \wedge Y \doteq Z) \Rightarrow X \doteq Z$

Proof:
analogous with
$\mathcal{E}_{\varphi[a/X][b/Y][c/Z]}(X) = \mathcal{E}_{\varphi[a/X][b/Y][c/Z]}(Y)$
and
$\mathcal{E}_{\varphi[a/X][b/Y][c/Z]}(Y) = \mathcal{E}_{\varphi[a/X][b/Y][c/Z]}(Z)$
implies
$\mathcal{E}_{\varphi[a/X][b/Y][c/Z]}(X) = \mathcal{E}_{\varphi[a/X][b/Y][c/Z]}(Z)$

Leibniz Equality in Σ-Models



$\doteq$ **is equivalence relation**

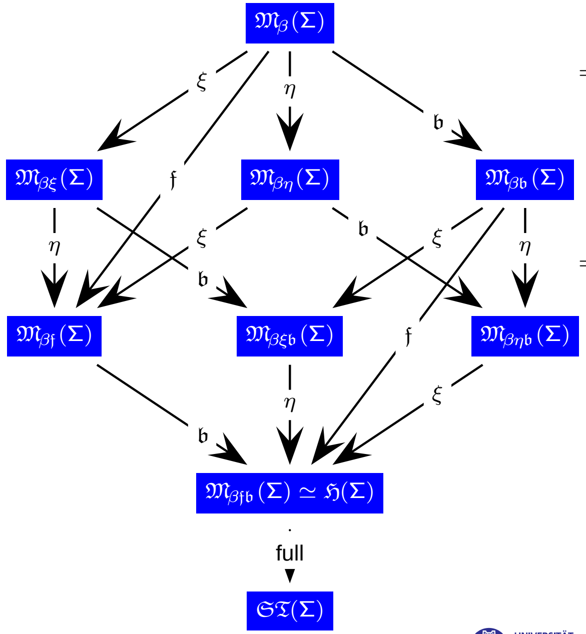
- $\forall X_\alpha.X \doteq X$
- $\forall X_\alpha, Y_\alpha.X \doteq Y \Rightarrow Y \doteq X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha.(X \doteq Y \wedge Y \doteq Z) \Rightarrow X \doteq Z$

$\doteq$ **is congruence relation**

- $\forall X_\alpha, Y_\alpha, F_{\alpha\to\beta}.X \doteq Y \Rightarrow (FX) \doteq (FY)$

Proof:
analogous with
$\mathcal{E}_{\varphi[a/X][b/Y][f/F]}(X) = \mathcal{E}_{\varphi[a/X][b/Y][f/F]}(Y)$
implies
$\mathcal{E}_{\varphi[a/X][b/Y][f/F]}(F)@\mathcal{E}_{\varphi[a/X][b/Y][f/F]}(X) = \mathcal{E}_{\varphi[a/X][b/Y][f/F]}(F)@\mathcal{E}_{\varphi[a/X][b/Y][f/F]}(Y)$

Leibniz Equality in Σ-Models



$\doteq$ **is equivalence relation**

- $\forall X_\alpha. X \doteq X$
- $\forall X_\alpha, Y_\alpha. X \doteq Y \Rightarrow Y \doteq X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha. (X \doteq Y \wedge Y \doteq Z) \Rightarrow X \doteq Z$

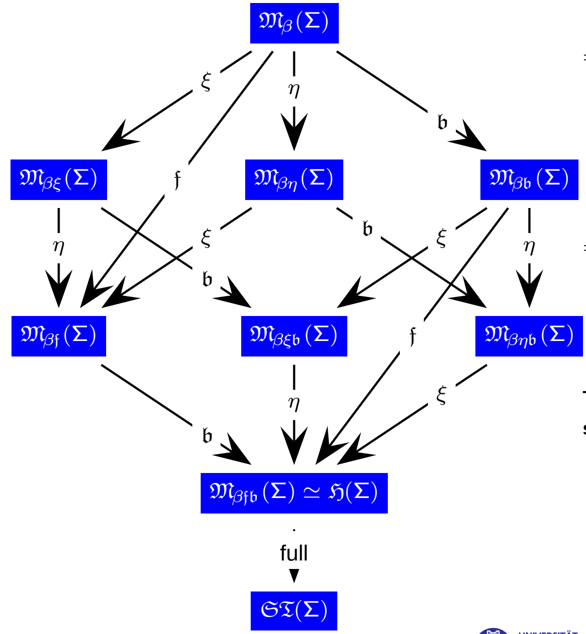
$\doteq$ **is congruence relation**

- $\forall X_\alpha, Y_\alpha, F_{\alpha\to\beta}. X \doteq Y \Rightarrow (FX) \doteq (FY)$
- $\forall X_\alpha, Y_\alpha, P_{\alpha\to o}. X \doteq Y \wedge (PX) \Rightarrow (PY)$

Proof:

analogous

Leibniz Equality in Σ-Models



$\doteq$ **is equivalence relation**

- $\forall X_\alpha. X \doteq X$
- $\forall X_\alpha, Y_\alpha. X \doteq Y \Rightarrow Y \doteq X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha. (X \doteq Y \wedge Y \doteq Z) \Rightarrow X \doteq Z$

$\doteq$ **is congruence relation**

- $\forall X_\alpha, Y_\alpha, F_{\alpha\to\beta}. X \doteq Y \Rightarrow (FX) \doteq (FY)$
- $\forall X_\alpha, Y_\alpha, P_{\alpha\to o}. X \doteq Y \wedge (PX) \Rightarrow (PY)$

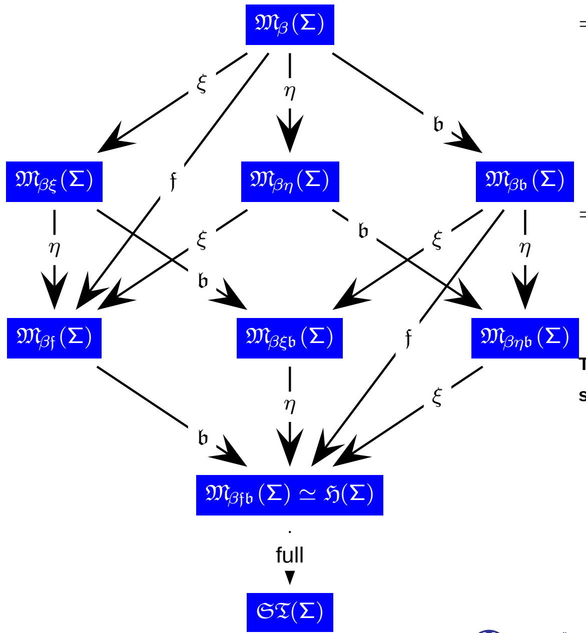
Trivial directions of Boolean and functional extensionality

- $\forall A_o, B_o. A \doteq B \Rightarrow (A \Leftrightarrow B)$

Proof:

by Theorem 'Trivial Extensionality Directions'

Leibniz Equality in Σ-Models



$\doteq$ **is equivalence relation**

- $\forall X_\alpha. X \doteq X$
- $\forall X_\alpha, Y_\alpha. X \doteq Y \Rightarrow Y \doteq X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha. (X \doteq Y \wedge Y \doteq Z) \Rightarrow X \doteq Z$

$\doteq$ **is congruence relation**

- $\forall X_\alpha, Y_\alpha, F_{\alpha\to\beta}. X \doteq Y \Rightarrow (FX) \doteq (FY)$
- $\forall X_\alpha, Y_\alpha, P_{\alpha\to o}. X \doteq Y \wedge (PX) \Rightarrow (PY)$

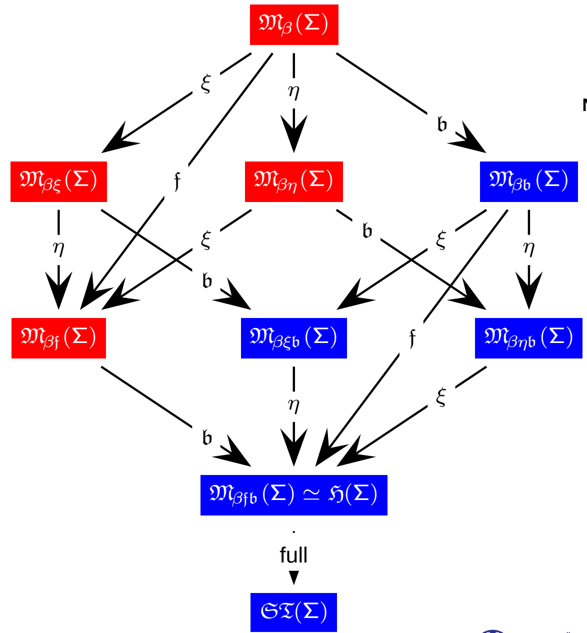
Trivial directions of Boolean and functional extensionality

- $\forall A_o, B_o. A \doteq B \Rightarrow (A \Leftrightarrow B)$
- $\forall F_{\alpha\to\beta}, G_{\alpha\to\beta}. F \doteq G \Rightarrow (\forall X_\alpha. FX \doteq GX)$

Proof:

by Theorem 'Trivial Extensionality Directions'

Leibniz Equality in Σ-Models



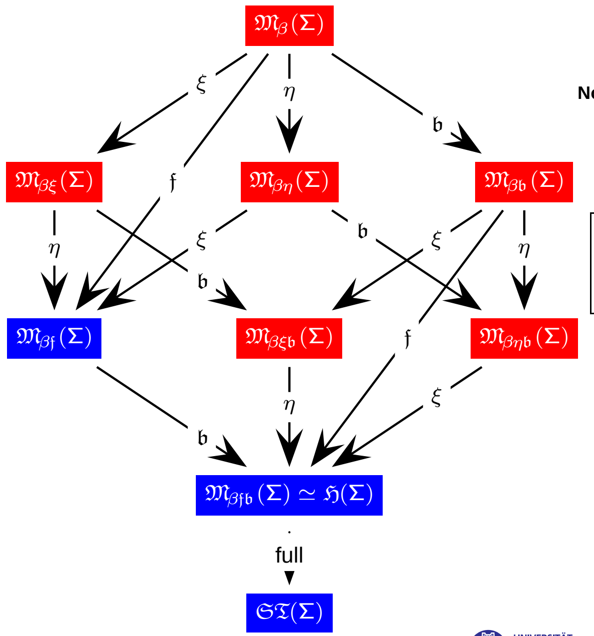
Non-trivial direction of Boolean extensionality

- $\forall A_o, B_o. (A \Leftrightarrow B) \Rightarrow A \doteq B$

Proof:

by Theorem 'Extensionality in Σ-Models'

Leibniz Equality in Σ-Models



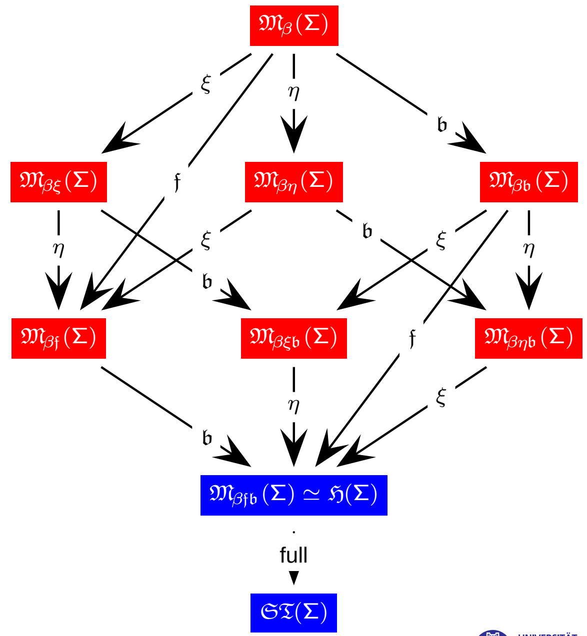
Non-trivial direct. of functional extensionality

- $\forall F_{\alpha\to\beta}, G_{\alpha\to\beta}. (\forall X_\alpha. FX \doteq GX) \Rightarrow F \doteq G$

Proof:

by Theorem 'Extensionality in Σ-Models'

Further Examples



- $\forall X. \forall Y. X \lor Y \Leftrightarrow Y \lor X$
- $\forall X. \forall Y. X \lor Y \doteq Y \lor X$
- $\lambda X. \lambda Y. X \lor Y \doteq \lambda X. \lambda Y. Y \lor X$
- $\lor \doteq \lambda X. \lambda Y. Y \lor X$

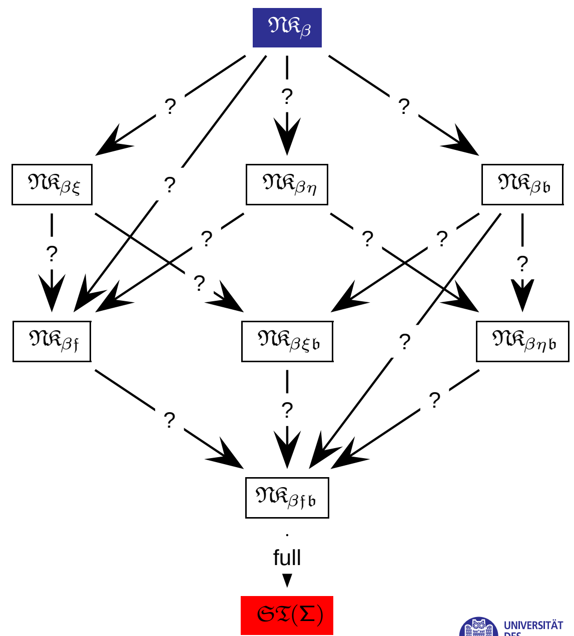
validity requires $\mathfrak{b}$ and $\mathfrak{f}$

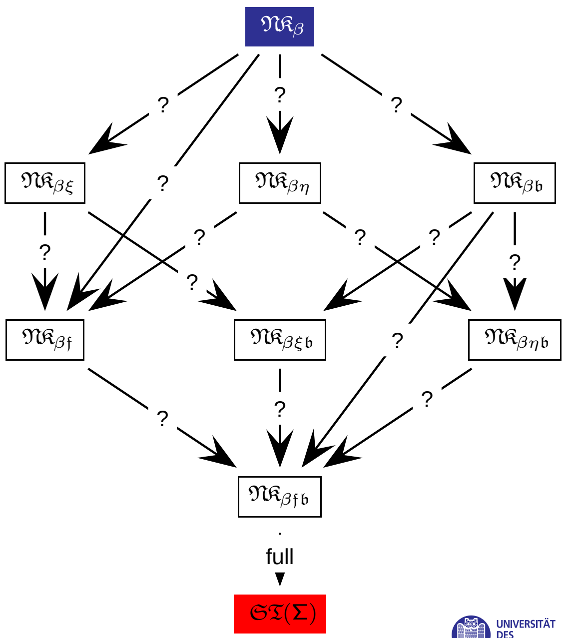
Proof:

Exercise

Calculi: ND for HOL

ND for HOL: Base Calculus $\mathfrak{NK}_\beta$





Base Calculus $\mathfrak{NK}_\beta$

$\overline{\quad}\ \mathfrak{NK}(Hyp)$ $\quad\overline{\quad}\ \mathfrak{NK}(\beta)$

$\overline{\quad}\ \mathfrak{NK}(\neg I)$ $\quad\overline{\quad}\ \mathfrak{NK}(\neg E)$

$\overline{\quad}\ \mathfrak{NK}(\vee I_L)$ $\quad\overline{\quad}\ \mathfrak{NK}(\vee I_R)$

$\overline{\quad}\ \mathfrak{NK}(\vee E)$

$\overline{\quad}\ \mathfrak{NK}(\Pi I)^w$

$\overline{\quad}\ \mathfrak{NK}(\Pi E)$ $\quad\overline{\quad}\ \mathfrak{NK}(Contr)$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\ \mathfrak{NK}(Hyp)$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\ \mathfrak{NK}(Hyp) \qquad \frac{\mathbf{A} =_\beta \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\ \mathfrak{NK}(\beta)$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\ \mathfrak{NK}(Hyp) \qquad \frac{\mathbf{A} =_\beta \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\ \mathfrak{NK}(\beta)$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg\mathbf{A}}\ \mathfrak{NK}(\neg I)$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

ND for HOL: Base Calculus $\mathfrak{NK}_\beta$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Hyp) \qquad \frac{\mathbf{A} =_\beta \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\beta)$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg \mathbf{A}}\,\mathfrak{NK}(\neg I) \qquad \frac{\Phi \Vdash \neg \mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\neg E)$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

ND for HOL: Base Calculus $\mathfrak{NK}_\beta$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Hyp) \qquad \frac{\mathbf{A} =_\beta \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\beta)$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg \mathbf{A}}\,\mathfrak{NK}(\neg I) \qquad \frac{\Phi \Vdash \neg \mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\neg E)$$

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \lor \mathbf{B}}\,\mathfrak{NK}(\lor I_L)$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[3] – p.158

ND for HOL: Base Calculus $\mathfrak{NK}_\beta$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Hyp) \qquad \frac{\mathbf{A} =_\beta \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\beta)$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg \mathbf{A}}\,\mathfrak{NK}(\neg I) \qquad \frac{\Phi \Vdash \neg \mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\neg E)$$

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \lor \mathbf{B}}\,\mathfrak{NK}(\lor I_L) \qquad \frac{\Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \lor \mathbf{B}}\,\mathfrak{NK}(\lor I_R)$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[3] – p.158

ND for HOL: Base Calculus $\mathfrak{NK}_\beta$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Hyp) \qquad \frac{\mathbf{A} =_\beta \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\beta)$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg \mathbf{A}}\,\mathfrak{NK}(\neg I) \qquad \frac{\Phi \Vdash \neg \mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\neg E)$$

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \lor \mathbf{B}}\,\mathfrak{NK}(\lor I_L) \qquad \frac{\Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \lor \mathbf{B}}\,\mathfrak{NK}(\lor I_R)$$

$$\frac{\Phi \Vdash \mathbf{A} \lor \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\lor E)$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[3] – p.158

ND for HOL: Base Calculus $\mathfrak{NK}_\beta$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Hyp) \qquad \frac{\mathbf{A} =_\beta \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\beta)$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg\mathbf{A}}\,\mathfrak{NK}(\neg I) \qquad \frac{\Phi \Vdash \neg\mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\neg E)$$

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \vee \mathbf{B}}\,\mathfrak{NK}(\vee I_L) \qquad \frac{\Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \vee \mathbf{B}}\,\mathfrak{NK}(\vee I_R)$$

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\vee E)$$

$$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w} \text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\,\mathfrak{NK}(\Pi I)^{\mathsf{w}}$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

ND for HOL: Base Calculus $\mathfrak{NK}_\beta$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Hyp) \qquad \frac{\mathbf{A} =_\beta \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\beta)$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg\mathbf{A}}\,\mathfrak{NK}(\neg I) \qquad \frac{\Phi \Vdash \neg\mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\neg E)$$

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \vee \mathbf{B}}\,\mathfrak{NK}(\vee I_L) \qquad \frac{\Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \vee \mathbf{B}}\,\mathfrak{NK}(\vee I_R)$$

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\vee E)$$

$$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w} \text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\,\mathfrak{NK}(\Pi I)^{\mathsf{w}}$$

$$\frac{\Phi \Vdash \Pi^\alpha \mathbf{G}}{\Phi \Vdash \mathbf{G}\mathbf{A}}\,\mathfrak{NK}(\Pi E)$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

ND for HOL: Base Calculus $\mathfrak{NK}_\beta$

$\mathfrak{NK}_\beta$:

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Hyp) \qquad \frac{\mathbf{A} =_\beta \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\beta)$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg\mathbf{A}}\,\mathfrak{NK}(\neg I) \qquad \frac{\Phi \Vdash \neg\mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\neg E)$$

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \vee \mathbf{B}}\,\mathfrak{NK}(\vee I_L) \qquad \frac{\Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \vee \mathbf{B}}\,\mathfrak{NK}(\vee I_R)$$

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\vee E)$$

$$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w} \text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\,\mathfrak{NK}(\Pi I)^{\mathsf{w}}$$

$$\frac{\Phi \Vdash \Pi^\alpha \mathbf{G}}{\Phi \Vdash \mathbf{G}\mathbf{A}}\,\mathfrak{NK}(\Pi E) \qquad \frac{\Phi * \neg\mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Contr)$$

$\Phi * \mathbf{A} := \Phi \cup \{\mathbf{A}\}$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \wedge \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\wedge E_L)$$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\land E_L) \qquad \frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\land E_R)$$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\land E_L) \qquad \frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\land E_R) \qquad \frac{\Phi \Vdash \mathbf{A} \quad \Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \land \mathbf{B}}\,\mathfrak{NK}(\land I)$$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\land E_L) \qquad \frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\land E_R) \qquad \frac{\Phi \Vdash \mathbf{A} \quad \Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \land \mathbf{B}}\,\mathfrak{NK}(\land I)$$

$$\frac{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\Rightarrow E)$$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\land E_L) \qquad \frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\land E_R) \qquad \frac{\Phi \Vdash \mathbf{A} \quad \Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \land \mathbf{B}}\,\mathfrak{NK}(\land I)$$

$$\frac{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\Rightarrow E) \qquad \frac{\Phi, \mathbf{A} \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B}}\,\mathfrak{NK}(\Rightarrow I)$$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\land E_L) \qquad \frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\land E_R) \qquad \frac{\Phi \Vdash \mathbf{A} \quad \Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \land \mathbf{B}}\,\mathfrak{NK}(\land I)$$

$$\frac{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\Rightarrow E) \qquad \frac{\Phi, \mathbf{A} \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B}}\,\mathfrak{NK}(\Rightarrow I)$$

$$\frac{\Phi \Vdash \mathbf{G}\mathbf{T}_\alpha}{\Phi \Vdash \Sigma^\alpha \mathbf{G}}\,\mathfrak{NK}(\Sigma I)$$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\land E_L) \qquad \frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\land E_R) \qquad \frac{\Phi \Vdash \mathbf{A} \quad \Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \land \mathbf{B}}\,\mathfrak{NK}(\land I)$$

$$\frac{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\Rightarrow E) \qquad \frac{\Phi, \mathbf{A} \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B}}\,\mathfrak{NK}(\Rightarrow I)$$

$$\frac{\Phi \Vdash \mathbf{G}\mathbf{T}_\alpha}{\Phi \Vdash \Sigma^\alpha \mathbf{G}}\,\mathfrak{NK}(\Sigma I) \qquad \frac{\Phi \Vdash \Sigma^\alpha \mathbf{G} \quad \Phi * \mathbf{G}\mathsf{w}_\alpha \Vdash \mathbf{C} \quad \mathsf{w}\text{ new parameter}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\Sigma E)$$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\land E_L) \qquad \frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\land E_R) \qquad \frac{\Phi \Vdash \mathbf{A} \quad \Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \land \mathbf{B}}\,\mathfrak{NK}(\land I)$$

$$\frac{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\Rightarrow E) \qquad \frac{\Phi, \mathbf{A} \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B}}\,\mathfrak{NK}(\Rightarrow I)$$

$$\frac{\Phi \Vdash \mathbf{G}\mathbf{T}_\alpha}{\Phi \Vdash \Sigma^\alpha \mathbf{G}}\,\mathfrak{NK}(\Sigma I) \qquad \frac{\Phi \Vdash \Sigma^\alpha \mathbf{G} \quad \Phi * \mathbf{G}\mathsf{w}_\alpha \Vdash \mathbf{C} \quad \mathsf{w}\text{ new parameter}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\Sigma E)$$

$$\frac{\Phi \Vdash \mathbf{T} =^\alpha \mathbf{W} \quad \Phi \Vdash \mathbf{A}[\mathbf{T}]}{\Phi \Vdash \mathbf{A}[\mathbf{W}]}\,\mathfrak{NK}(=\mathit{Subst})$$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\land E_L) \qquad \frac{\Phi \Vdash \mathbf{A} \land \mathbf{B}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\land E_R) \qquad \frac{\Phi \Vdash \mathbf{A} \quad \Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \land \mathbf{B}}\,\mathfrak{NK}(\land I)$$

$$\frac{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\Rightarrow E) \qquad \frac{\Phi, \mathbf{A} \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B}}\,\mathfrak{NK}(\Rightarrow I)$$

$$\frac{\Phi \Vdash \mathbf{G}\mathbf{T}_\alpha}{\Phi \Vdash \Sigma^\alpha \mathbf{G}}\,\mathfrak{NK}(\Sigma I) \qquad \frac{\Phi \Vdash \Sigma^\alpha \mathbf{G} \quad \Phi * \mathbf{G}\mathsf{w}_\alpha \Vdash \mathbf{C} \quad \mathsf{w}\text{ new parameter}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\Sigma E)$$

$$\frac{\Phi \Vdash \mathbf{T} =^\alpha \mathbf{W} \quad \Phi \Vdash \mathbf{A}[\mathbf{T}]}{\Phi \Vdash \mathbf{A}[\mathbf{W}]}\,\mathfrak{NK}(=\mathit{Subst}) \qquad \frac{}{\Phi \Vdash \mathbf{A} = \mathbf{A}}\,\mathfrak{NK}(=\mathit{Refl})$$

ND for HOL: Rules for Richer Signatures

Inference rules for $\mathfrak{NK}_\beta$ (for richer signatures)

$$\frac{\Phi \Vdash \mathbf{A} \wedge \mathbf{B}}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\wedge E_L) \quad \frac{\Phi \Vdash \mathbf{A} \wedge \mathbf{B}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\wedge E_R) \quad \frac{\Phi \Vdash \mathbf{A} \quad \Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \wedge \mathbf{B}}\,\mathfrak{NK}(\wedge I)$$

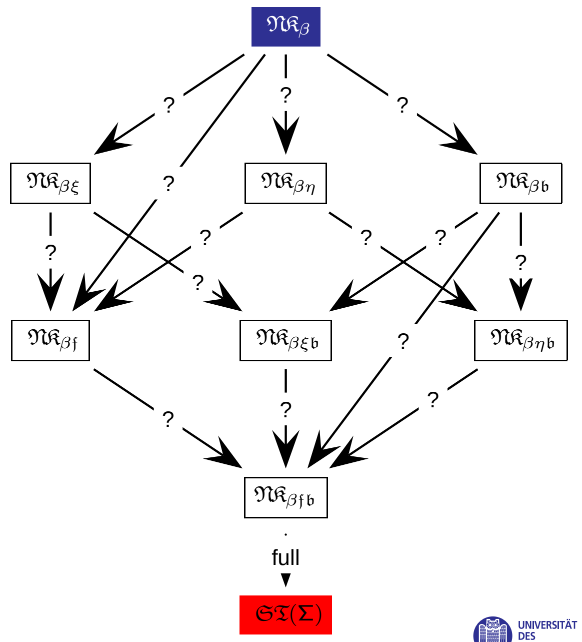
$$\frac{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\Rightarrow E) \quad \frac{\Phi, \mathbf{A} \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \Rightarrow \mathbf{B}}\,\mathfrak{NK}(\Rightarrow I)$$

$$\frac{\Phi \Vdash \mathbf{G}\mathbf{T}_\alpha}{\Phi \Vdash \Sigma^\alpha \mathbf{G}}\,\mathfrak{NK}(\Sigma I) \quad \frac{\Phi \Vdash \Sigma^\alpha \mathbf{G} \quad \Phi * \mathbf{G}\mathrm{w}_\alpha \Vdash \mathbf{C} \quad \text{w new parameter}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\Sigma E)$$

$$\frac{\Phi \Vdash \mathbf{T} =^\alpha \mathbf{W} \quad \Phi \Vdash \mathbf{A}[\mathbf{T}]}{\Phi \Vdash \mathbf{A}[\mathbf{W}]}\,\mathfrak{NK}(= Subst) \quad \frac{}{\Phi \Vdash \mathbf{A} = \mathbf{A}}\,\mathfrak{NK}(= Refl)$$

Here: we define logical constants $\wedge, \Rightarrow, \Sigma$, etc. in terms of $\neg, \vee, \Pi$ as usual and strictly use Leibniz equality instead of primitive equality; then the above rules are not needed

ND for HOL: Extensionality Rules



Base Calculus $\mathfrak{NK}_\beta$

$\mathfrak{NK}(Hyp)$ $\mathfrak{NK}(\beta)$

$\mathfrak{NK}(\neg I)$ $\mathfrak{NK}(\neg E)$

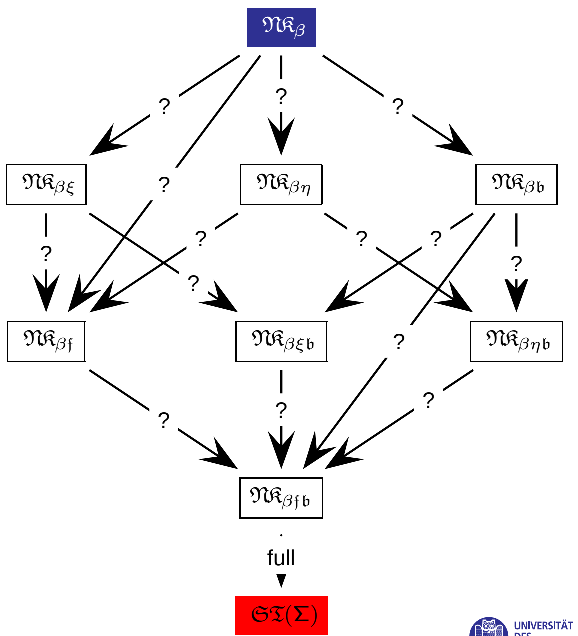
$\mathfrak{NK}(\vee I_L)$ $\mathfrak{NK}(\vee I_R)$

$\mathfrak{NK}(\vee E)$

$\mathfrak{NK}(\Pi I)^{\mathrm{w}}$

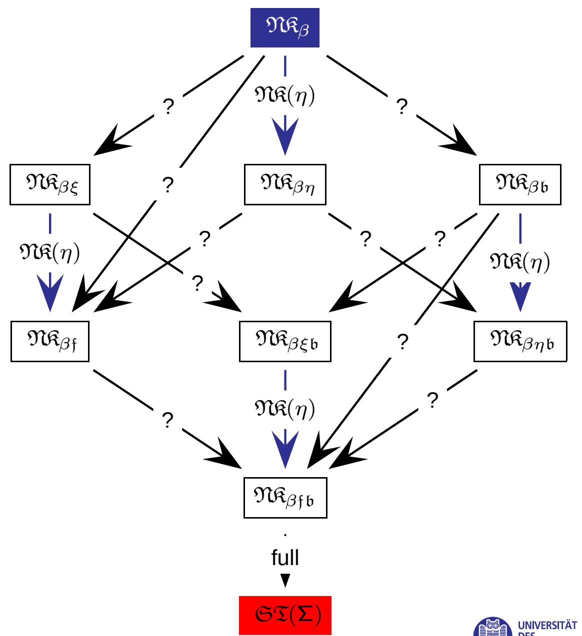
$\mathfrak{NK}(\Pi E)$ $\mathfrak{NK}(Contr)$

ND for HOL: Extensionality Rules



Optional Extensionality Rules

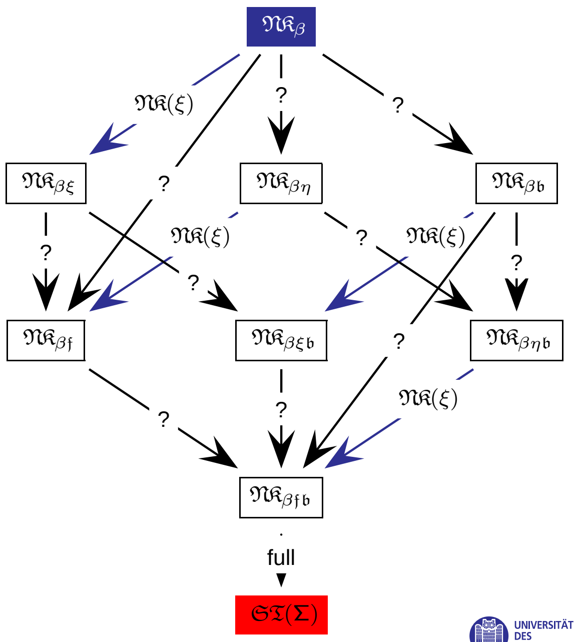
ND for HOL: Extensionality Rules



Optional Extensionality Rules

$$\frac{\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\eta)$$

ND for HOL: Extensionality Rules

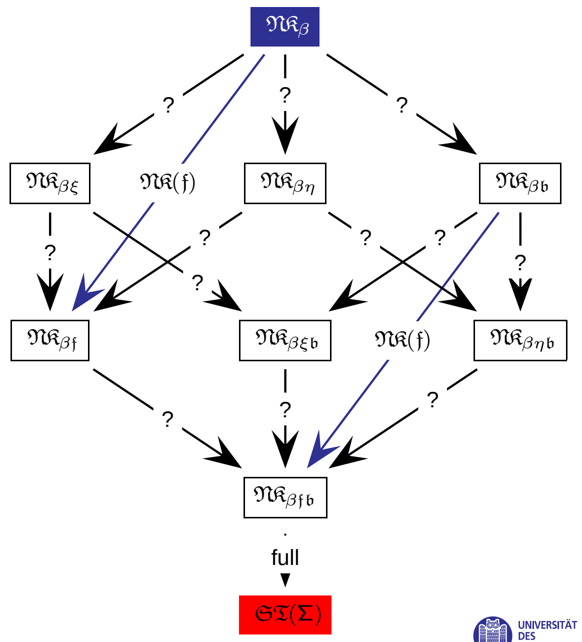


Optional Extensionality Rules

$$\frac{A \overset{\beta\eta}{=} B \quad \Phi \Vdash A}{\Phi \Vdash B}\,\mathfrak{NK}(\eta)$$

$$\frac{\Phi \Vdash \forall X_\alpha . M \doteq^{\beta} N}{\Phi \Vdash (\lambda X_\alpha . M) \doteq^{\beta\alpha} (\lambda X_\alpha . N)}\,\mathfrak{NK}(\xi)$$

ND for HOL: Extensionality Rules



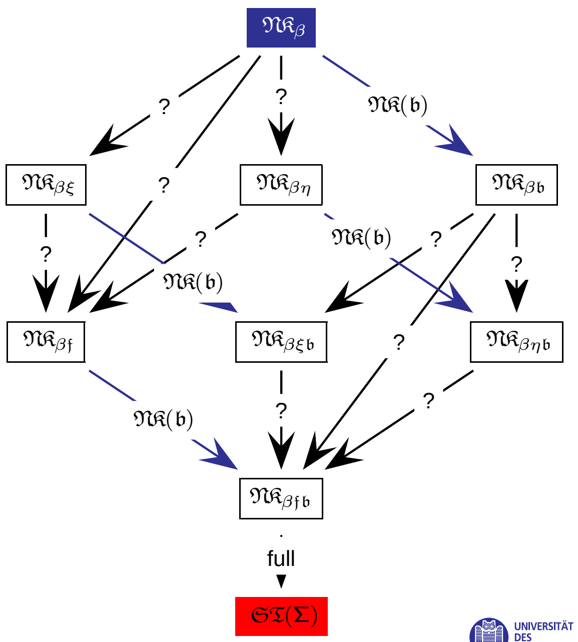
Optional Extensionality Rules

$$\frac{A \overset{\beta\eta}{=} B \quad \Phi \Vdash A}{\Phi \Vdash B}\,\mathfrak{NK}(\eta)$$

$$\frac{\Phi \Vdash \forall X_\alpha . M \doteq^{\beta} N}{\Phi \Vdash (\lambda X_\alpha . M) \doteq^{\beta\alpha} (\lambda X_\alpha . N)}\,\mathfrak{NK}(\xi)$$

$$\frac{\Phi \Vdash \forall X_\alpha . G X \doteq^{\beta} H X}{\Phi \Vdash G \doteq^{\beta\alpha} H}\,\mathfrak{NK}(\mathfrak{f})$$

ND for HOL: Extensionality Rules



Optional Extensionality Rules

$$\frac{A \overset{\beta\eta}{=} B \quad \Phi \Vdash A}{\Phi \Vdash B}\,\mathfrak{NK}(\eta)$$

$$\frac{\Phi \Vdash \forall X_\alpha . M \doteq^{\beta} N}{\Phi \Vdash (\lambda X_\alpha . M) \doteq^{\beta\alpha} (\lambda X_\alpha . N)}\,\mathfrak{NK}(\xi)$$

$$\frac{\Phi \Vdash \forall X_\alpha . G X \doteq^{\beta} H X}{\Phi \Vdash G \doteq^{\beta\alpha} H}\,\mathfrak{NK}(\mathfrak{f})$$

$$\frac{\Phi * A \Vdash B \quad \Phi * B \Vdash A}{\Phi \Vdash A \doteq^{o} B}\,\mathfrak{NK}(\mathfrak{b})$$

ND Calculi for HOL

Defn.: The Calculi $\mathfrak{NK}_*$

ND Calculi for HOL

Defn.: The Calculi $\mathfrak{NK}_*$

- The calculus $\mathfrak{NK}_\beta$ consists of the inference rules for $\mathfrak{NK}_\beta$ for the provability judgment $\Vdash$ between sets of sentences Φ and sentences $\mathbf{A}$.

ND Calculi for HOL

Defn.: The Calculi $\mathfrak{NK}_*$

- The calculus $\mathfrak{NK}_\beta$ consists of the inference rules for $\mathfrak{NK}_\beta$ for the provability judgment $\Vdash$ between sets of sentences Φ and sentences $\mathbf{A}$.
- We write $\Vdash \mathbf{A}$ for $\emptyset \Vdash \mathbf{A}$.

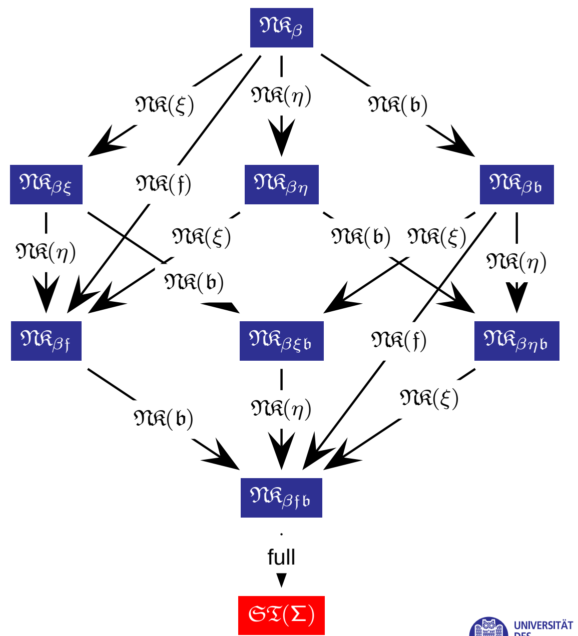
ND Calculi for HOL

Defn.: The Calculi $\mathfrak{NK}_*$

- The calculus $\mathfrak{NK}_\beta$ consists of the inference rules for $\mathfrak{NK}_\beta$ for the provability judgment $\Vdash$ between sets of sentences Φ and sentences $\mathbf{A}$.
- We write $\Vdash \mathbf{A}$ for $\emptyset \Vdash \mathbf{A}$.
- For $* \in \{\beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ we obtain the calculus $\mathfrak{NK}_*$ by adding the respective extensionality rules when specified in $*$:

$$\mathfrak{NK}_{\beta\eta}, \mathfrak{NK}_{\beta\xi}, \mathfrak{NK}_{\beta\mathfrak{f}}, \mathfrak{NK}_{\beta\mathfrak{b}}, \mathfrak{NK}_{\beta\eta\mathfrak{b}}, \mathfrak{NK}_{\beta\xi\mathfrak{b}}, \mathfrak{NK}_{\beta\mathfrak{f}\mathfrak{b}}$$

ND for HOL: The Complete Picture



Base Calculus $\mathfrak{NK}_\beta$

— $\mathfrak{NK}(Hyp)$ — $\mathfrak{NK}(\beta)$

— $\mathfrak{NK}(\neg I)$ — $\mathfrak{NK}(\neg E)$

— $\mathfrak{NK}(\vee I_L)$ — $\mathfrak{NK}(\vee I_R)$

— $\mathfrak{NK}(\vee E)$

— $\mathfrak{NK}(\Pi I)^w$

— $\mathfrak{NK}(\Pi E)$ — $\mathfrak{NK}(Contr)$

Optional Extensionality Rules

— $\mathfrak{NK}(\eta)$ — $\mathfrak{NK}(\xi)$

— $\mathfrak{NK}(\mathfrak{f})$ — $\mathfrak{NK}(\mathfrak{b})$

ND Example Proof in $\mathfrak{NK}_\beta$

Derivation of:

$$(\mathsf{A} \doteq^\alpha \mathsf{A})$$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{}{\Vdash_{\mathfrak{NK}_\beta} (\mathsf{A} \doteq^\alpha \mathsf{A}) := \Pi^\alpha(\lambda \mathsf{P}(\neg(\mathsf{PA}) \vee (\mathsf{PA})))}\,\mathfrak{NK}(III)$$

Derivation of:

$$(\mathsf{A} \doteq^\alpha \mathsf{A})$$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{}{\Vdash_{\mathfrak{NK}_\beta} ((\lambda \mathsf{P}(\neg(\mathsf{PA}) \vee (\mathsf{PA})))\mathsf{q})}\,\mathfrak{NK}(\beta)}{\Vdash_{\mathfrak{NK}_\beta} (\mathsf{A} \doteq^\alpha \mathsf{A}) := \Pi^\alpha(\lambda \mathsf{P}(\neg(\mathsf{PA}) \vee (\mathsf{PA})))}\,\mathfrak{NK}(III)$$

Derivation of:

$$(\mathsf{A} \doteq^\alpha \mathsf{A})$$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{}{\Vdash_{\mathfrak{NK}_\beta} \neg(\mathsf{qA}) \vee (\mathsf{qA})}\,\mathfrak{NK}(Contr)}{\Vdash_{\mathfrak{NK}_\beta} ((\lambda \mathsf{P}(\neg(\mathsf{PA}) \vee (\mathsf{PA})))\mathsf{q})}\,\mathfrak{NK}(\beta)}{\Vdash_{\mathfrak{NK}_\beta} (\mathsf{A} \doteq^\alpha \mathsf{A}) := \Pi^\alpha(\lambda \mathsf{P}(\neg(\mathsf{PA}) \vee (\mathsf{PA})))}\,\mathfrak{NK}(III)$$

Derivation of:

$$(\mathsf{A} \doteq^\alpha \mathsf{A})$$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Phi^1 := \{\neg(\neg(qA) \vee (qA))\} \Vdash_{\mathfrak{NK}_\beta} \mathbf{F}_o}\;\mathfrak{NK}(\neg E)}{\Vdash_{\mathfrak{NK}_\beta} \neg(qA) \vee (qA)}\;\mathfrak{NK}(Contr)}{\Vdash_{\mathfrak{NK}_\beta} ((\lambda P(\neg(PA) \vee (PA)))q)}\;\mathfrak{NK}(\beta)}{\Vdash_{\mathfrak{NK}_\beta} (A \doteq^\alpha A) := \Pi^\alpha(\lambda P(\neg(PA) \vee (PA)))}\;\mathfrak{NK}(\Pi I)$$

Derivation of:

$$(A \doteq^\alpha A)$$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{\dfrac{\Phi^1 \Vdash_{\mathfrak{NK}_\beta} \neg(qA) \vee (qA)}{\Phi^1 := \{\neg(\neg(qA) \vee (qA))\} \Vdash_{\mathfrak{NK}_\beta} \mathbf{F}_o}\;\mathfrak{NK}(\neg E)}{\Vdash_{\mathfrak{NK}_\beta} \neg(qA) \vee (qA)}\;\mathfrak{NK}(Contr)}{\Vdash_{\mathfrak{NK}_\beta} ((\lambda P(\neg(PA) \vee (PA)))q)}\;\mathfrak{NK}(\beta)}{\Vdash_{\mathfrak{NK}_\beta} (A \doteq^\alpha A) := \Pi^\alpha(\lambda P(\neg(PA) \vee (PA)))}\;\mathfrak{NK}(\Pi I)$$

Derivation of:

$$(A \doteq^\alpha A)$$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{\dfrac{\Phi^1 \Vdash_{\mathfrak{NK}_\beta} \neg(qA) \vee (qA) \qquad \dfrac{}{\Phi^1 \Vdash_{\mathfrak{NK}_\beta} \{\neg(\neg(qA) \vee (qA))\}}\;\mathfrak{NK}(Hyp)}{\Phi^1 := \{\neg(\neg(qA) \vee (qA))\} \Vdash_{\mathfrak{NK}_\beta} \mathbf{F}_o}\;\mathfrak{NK}(\neg E)}{\Vdash_{\mathfrak{NK}_\beta} \neg(qA) \vee (qA)}\;\mathfrak{NK}(Contr)}{\Vdash_{\mathfrak{NK}_\beta} ((\lambda P(\neg(PA) \vee (PA)))q)}\;\mathfrak{NK}(\beta)}{\Vdash_{\mathfrak{NK}_\beta} (A \doteq^\alpha A) := \Pi^\alpha(\lambda P(\neg(PA) \vee (PA)))}\;\mathfrak{NK}(\Pi I)$$

Derivation of:

$$(A \doteq^\alpha A)$$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{\dfrac{\Phi^1 \Vdash_{\mathfrak{NK}_\beta} \neg(qA) \vee (qA) \qquad \dfrac{}{\Phi^1 \Vdash_{\mathfrak{NK}_\beta} \{\neg(\neg(qA) \vee (qA))\}}\;\mathfrak{NK}(Hyp)}{\Phi^1 := \{\neg(\neg(qA) \vee (qA))\} \Vdash_{\mathfrak{NK}_\beta} \mathbf{F}_o}\;\mathfrak{NK}(\neg E)}{\Vdash_{\mathfrak{NK}_\beta} \neg(qA) \vee (qA)}\;\mathfrak{NK}(Contr)}{\Vdash_{\mathfrak{NK}_\beta} ((\lambda P(\neg(PA) \vee (PA)))q)}\;\mathfrak{NK}(\beta)}{\Vdash_{\mathfrak{NK}_\beta} (A \doteq^\alpha A) := \Pi^\alpha(\lambda P(\neg(PA) \vee (PA)))}\;\mathfrak{NK}(\Pi I)$$

Derivation of:

$$(A \doteq^\alpha A)$$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{\dfrac{\overset{\text{See Next Slide}}{\color{red}{\Phi^1 \Vdash_{\mathfrak{NK}_\beta} \neg(\mathsf{qA}) \lor (\mathsf{qA})}} \qquad \dfrac{}{\Phi^1 \Vdash_{\mathfrak{NK}_\beta} \{\neg(\neg(\mathsf{qA}) \lor (\mathsf{qA}))}\,\mathfrak{NK}(Hyp)}{\Phi^1 := \{\neg(\neg(\mathsf{qA}) \lor (\mathsf{qA}))\} \Vdash_{\mathfrak{NK}_\beta} \mathbf{F}_o}\,\mathfrak{NK}(\neg E)}{\Vdash_{\mathfrak{NK}_\beta} \neg(\mathsf{qA}) \lor (\mathsf{qA})}\,\mathfrak{NK}(Contr)}{\Vdash_{\mathfrak{NK}_\beta} ((\lambda \mathsf{P}(\neg(\mathsf{PA}) \lor (\mathsf{PA})))\mathsf{q})}\,\mathfrak{NK}(\beta)}{\Vdash_{\mathfrak{NK}_\beta} (\mathsf{A} \doteq^\alpha \mathsf{A}) := \Pi^\alpha(\lambda \mathsf{P}(\neg(\mathsf{PA}) \lor (\mathsf{PA})))}\,\mathfrak{NK}(III)$$

Derivation of:

$$(\mathsf{A} \doteq^\alpha \mathsf{A})$$

ND Example Proof in $\mathfrak{NK}_\beta$

Derivation of:

$$\{\neg(\neg \mathsf{p} \lor \mathsf{p})\} \Vdash_{\mathfrak{NK}_\beta} \neg \mathsf{p} \lor \mathsf{p}$$

resp. $\color{red}{\{\neg(\neg(\mathsf{qA}) \lor (\mathsf{qA}))\} \Vdash_{\mathfrak{NK}_\beta} \neg(\mathsf{qA}) \lor (\mathsf{qA})}$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{}{\{\neg(\neg \mathsf{p} \lor \mathsf{p})\} \Vdash_{\mathfrak{NK}_\beta} \neg \mathsf{p} \lor \mathsf{p}}\,\mathfrak{NK}(\lor I_L)$$

Derivation of:

$$\{\neg(\neg \mathsf{p} \lor \mathsf{p})\} \Vdash_{\mathfrak{NK}_\beta} \neg \mathsf{p} \lor \mathsf{p}$$

resp. $\color{red}{\{\neg(\neg(\mathsf{qA}) \lor (\mathsf{qA}))\} \Vdash_{\mathfrak{NK}_\beta} \neg(\mathsf{qA}) \lor (\mathsf{qA})}$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{}{\{\neg(\neg \mathsf{p} \lor \mathsf{p})\} \Vdash_{\mathfrak{NK}_\beta} \neg \mathsf{p}}\,\mathfrak{NK}(\neg I)}{\{\neg(\neg \mathsf{p} \lor \mathsf{p})\} \Vdash_{\mathfrak{NK}_\beta} \neg \mathsf{p} \lor \mathsf{p}}\,\mathfrak{NK}(\lor I_L)$$

Derivation of:

$$\{\neg(\neg \mathsf{p} \lor \mathsf{p})\} \Vdash_{\mathfrak{NK}_\beta} \neg \mathsf{p} \lor \mathsf{p}$$

resp. $\color{red}{\{\neg(\neg(\mathsf{qA}) \lor (\mathsf{qA}))\} \Vdash_{\mathfrak{NK}_\beta} \neg(\mathsf{qA}) \lor (\mathsf{qA})}$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{}{\Phi^2 := \{\neg(\neg p \lor p), p\} \Vdash_{\mathfrak{NK}_\beta} \mathbf{F}_o}\ \mathfrak{NK}(\neg E)}{\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p}\ \mathfrak{NK}(\neg I)}{\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p}\ \mathfrak{NK}(\lor I_L)$$

Derivation of:

$$\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p$$

resp. $\{\neg(\neg(qA) \lor (qA))\} \Vdash_{\mathfrak{NK}_\beta} \neg(qA) \lor (qA)$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Phi^2 \Vdash_{\mathfrak{NK}_\beta} \neg(\neg p \lor p)}\ \mathfrak{NK}(Hyp)}{\Phi^2 := \{\neg(\neg p \lor p), p\} \Vdash_{\mathfrak{NK}_\beta} \mathbf{F}_o}\ \mathfrak{NK}(\neg E)}{\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p}\ \mathfrak{NK}(\neg I)}{\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p}\ \mathfrak{NK}(\lor I_L)$$

Derivation of:

$$\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p$$

resp. $\{\neg(\neg(qA) \lor (qA))\} \Vdash_{\mathfrak{NK}_\beta} \neg(qA) \lor (qA)$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Phi^2 \Vdash_{\mathfrak{NK}_\beta} \neg(\neg p \lor p)}\ \mathfrak{NK}(Hyp) \qquad \dfrac{}{\Phi^2 \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p}\ \mathfrak{NK}(\lor I_R)}{\Phi^2 := \{\neg(\neg p \lor p), p\} \Vdash_{\mathfrak{NK}_\beta} \mathbf{F}_o}\ \mathfrak{NK}(\neg E)}{\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p}\ \mathfrak{NK}(\neg I)}{\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p}\ \mathfrak{NK}(\lor I_L)$$

Derivation of:

$$\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p$$

resp. $\{\neg(\neg(qA) \lor (qA))\} \Vdash_{\mathfrak{NK}_\beta} \neg(qA) \lor (qA)$

ND Example Proof in $\mathfrak{NK}_\beta$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Phi^2 \Vdash_{\mathfrak{NK}_\beta} \neg(\neg p \lor p)}\ \mathfrak{NK}(Hyp) \qquad \dfrac{\dfrac{}{\Phi^2 \Vdash_{\mathfrak{NK}_\beta} p}\ \mathfrak{NK}(Hyp)}{\Phi^2 \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p}\ \mathfrak{NK}(\lor I_R)}{\Phi^2 := \{\neg(\neg p \lor p), p\} \Vdash_{\mathfrak{NK}_\beta} \mathbf{F}_o}\ \mathfrak{NK}(\neg E)}{\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p}\ \mathfrak{NK}(\neg I)}{\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p}\ \mathfrak{NK}(\lor I_L)$$

Derivation of:

$$\{\neg(\neg p \lor p)\} \Vdash_{\mathfrak{NK}_\beta} \neg p \lor p$$

resp. $\{\neg(\neg(qA) \lor (qA))\} \Vdash_{\mathfrak{NK}_\beta} \neg(qA) \lor (qA)$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$.
That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$.
That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: By induction on the derivation of $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$.
That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (base case)

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}} \mathfrak{NK}(Hyp)$$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$.
That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (base case)

$$\frac{\mathbf{C} \in \Phi}{\Phi \Vdash \mathbf{C}} \mathfrak{NK}(Hyp)$$

$\mathcal{M} \models \mathbf{C}$ whenever $\mathcal{M} \models \Phi$ and $\mathbf{C} \in \Phi$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\mathbf{A} =_\beta \mathbf{C} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\, \mathfrak{NK}(\beta)$$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\mathbf{A} =_\beta \mathbf{C} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\, \mathfrak{NK}(\beta)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \mathbf{A}$ and $\mathbf{A} =_\beta \mathbf{C}$. Let $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ be a model of Φ.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\mathbf{A} =_\beta \mathbf{C} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\, \mathfrak{NK}(\beta)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \mathbf{A}$ and $\mathbf{A} =_\beta \mathbf{C}$. Let $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. By induction, we know $\mathcal{M} \models \mathbf{A}$ and so $\mathcal{M} \models \mathbf{C}$ since Σ-evaluations respect β-equality.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi * \neg\mathbf{C} \Vdash \mathbf{F}_o}{\Phi \Vdash \mathbf{C}}\, \mathfrak{NK}(Contr)$$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi * \neg\mathbf{C} \Vdash \mathbf{F}_o}{\Phi \Vdash \mathbf{C}}\ \mathfrak{NK}(Contr)$$

Suppose $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$, $\mathcal{M} \models \Phi$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi * \neg\mathbf{C} \Vdash \mathbf{F}_o$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi * \neg\mathbf{C} \Vdash \mathbf{F}_o}{\Phi \Vdash \mathbf{C}}\ \mathfrak{NK}(Contr)$$

Suppose $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$, $\mathcal{M} \models \Phi$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi * \neg\mathbf{C} \Vdash \mathbf{F}_o$. By a previous Lemma, $\mathcal{M} \not\models \mathbf{F}_o$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi * \neg\mathbf{C} \Vdash \mathbf{F}_o}{\Phi \Vdash \mathbf{C}}\ \mathfrak{NK}(Contr)$$

Suppose $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$, $\mathcal{M} \models \Phi$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi * \neg\mathbf{C} \Vdash \mathbf{F}_o$. By a previous Lemma, $\mathcal{M} \not\models \mathbf{F}_o$. So, we must have $\mathcal{M} \not\models \neg\mathbf{C}$ and, hence, $\mathcal{M} \models \mathbf{C}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg\mathbf{A}}\ \mathfrak{NK}(\neg I)$$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg \mathbf{A}} \mathfrak{NK}(\neg I)$$

Analogous to $\mathfrak{NK}(Contr)$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \neg \mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}} \mathfrak{NK}(\neg E)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \neg \mathbf{A}$ and $\Phi \Vdash \mathbf{A}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \neg \mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}} \mathfrak{NK}(\neg E)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \neg \mathbf{A}$ and $\Phi \Vdash \mathbf{A}$. By induction, any model in $\mathfrak{M}_*(\Sigma)$ of Φ would have to model both $\mathbf{A}$ and $\neg \mathbf{A}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \neg \mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}} \mathfrak{NK}(\neg E)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \neg \mathbf{A}$ and $\Phi \Vdash \mathbf{A}$. By induction, any model in $\mathfrak{M}_*(\Sigma)$ of Φ would have to model both $\mathbf{A}$ and $\neg \mathbf{A}$.
So, there is no such model of Φ and we are done.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \lor \mathbf{B}}\,\mathfrak{NK}(\lor I_L)$$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \lor \mathbf{B}}\,\mathfrak{NK}(\lor I_L)$$

Suppose $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$, $\mathcal{M} \models \Phi$, and $\Phi \Vdash (\mathbf{A} \lor \mathbf{B})$ follows from $\Phi \Vdash \mathbf{A}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \lor \mathbf{B}}\,\mathfrak{NK}(\lor I_L)$$

Suppose $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$, $\mathcal{M} \models \Phi$, and $\Phi \Vdash (\mathbf{A} \lor \mathbf{B})$ follows from
$\Phi \Vdash \mathbf{A}$. By induction, $\mathcal{M} \models \mathbf{A}$ and so $\mathcal{M} \models (\mathbf{A} \lor \mathbf{B})$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \lor \mathbf{B}}\,\mathfrak{NK}(\lor I_R)$$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \vee \mathbf{B}}\ \mathfrak{NK}(\vee I_R)$$

Analogous to $\mathfrak{NK}(\vee I_L)$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}}\ \mathfrak{NK}(\vee E)$$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}}\ \mathfrak{NK}(\vee E)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\mathbf{A} \vee \mathbf{B})$, $\Phi * \mathbf{A} \Vdash \mathbf{C}$ and $\Phi * \mathbf{B} \Vdash \mathbf{C}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}}\ \mathfrak{NK}(\vee E)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\mathbf{A} \vee \mathbf{B})$, $\Phi * \mathbf{A} \Vdash \mathbf{C}$ and $\Phi * \mathbf{B} \Vdash \mathbf{C}$. Let $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ be a model of Φ.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}} \mathfrak{NK}(\vee E)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\mathbf{A} \vee \mathbf{B})$, $\Phi * \mathbf{A} \Vdash \mathbf{C}$ and $\Phi * \mathbf{B} \Vdash \mathbf{C}$. Let $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. By induction, $\mathcal{M} \models \mathbf{A} \vee \mathbf{B}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}} \mathfrak{NK}(\vee E)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\mathbf{A} \vee \mathbf{B})$, $\Phi * \mathbf{A} \Vdash \mathbf{C}$ and $\Phi * \mathbf{B} \Vdash \mathbf{C}$. Let $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. By induction, $\mathcal{M} \models \mathbf{A} \vee \mathbf{B}$. If $\mathcal{M} \models \mathbf{A}$, then by induction $\mathcal{M} \models \mathbf{C}$ since $\Phi * \mathbf{A} \Vdash \mathbf{C}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}} \mathfrak{NK}(\vee E)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\mathbf{A} \vee \mathbf{B})$, $\Phi * \mathbf{A} \Vdash \mathbf{C}$ and $\Phi * \mathbf{B} \Vdash \mathbf{C}$. Let $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. By induction, $\mathcal{M} \models \mathbf{A} \vee \mathbf{B}$. If $\mathcal{M} \models \mathbf{A}$, then by induction $\mathcal{M} \models \mathbf{C}$ since $\Phi * \mathbf{A} \Vdash \mathbf{C}$. If $\mathcal{M} \models \mathbf{B}$, then by induction $\mathcal{M} \models \mathbf{C}$ since $\Phi * \mathbf{B} \Vdash \mathbf{C}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}} \mathfrak{NK}(\vee E)$$

Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\mathbf{A} \vee \mathbf{B})$, $\Phi * \mathbf{A} \Vdash \mathbf{C}$ and $\Phi * \mathbf{B} \Vdash \mathbf{C}$. Let $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. By induction, $\mathcal{M} \models \mathbf{A} \vee \mathbf{B}$. If $\mathcal{M} \models \mathbf{A}$, then by induction $\mathcal{M} \models \mathbf{C}$ since $\Phi * \mathbf{A} \Vdash \mathbf{C}$. If $\mathcal{M} \models \mathbf{B}$, then by induction $\mathcal{M} \models \mathbf{C}$ since $\Phi * \mathbf{B} \Vdash \mathbf{C}$. In either case, $\Phi \Vdash \mathbf{C}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w} \text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\ \mathfrak{NK}(\Pi)^{\mathsf{w}}$$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w} \text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\ \mathfrak{NK}(\Pi)^{\mathsf{w}}$$

Suppose $\Phi \Vdash (\Pi^\alpha \mathbf{G})$ follows from $\Phi \Vdash \mathbf{G}\mathsf{w}$ where w_α is a fresh parameter.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w} \text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\ \mathfrak{NK}(\Pi)^{\mathsf{w}}$$

Suppose $\Phi \Vdash (\Pi^\alpha \mathbf{G})$ follows from $\Phi \Vdash \mathbf{G}\mathsf{w}$ where w_α is a fresh parameter. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w} \text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\ \mathfrak{NK}(\Pi)^{\mathsf{w}}$$

Suppose $\Phi \Vdash (\Pi^\alpha \mathbf{G})$ follows from $\Phi \Vdash \mathbf{G}\mathsf{w}$ where w_α is a fresh parameter. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. Assume $\mathcal{M} \not\models \Pi^\alpha \mathbf{G}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases) $$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w}\text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\,\mathfrak{NK}(\Pi)^{\mathsf{w}}$$

Suppose $\Phi \Vdash (\Pi^\alpha \mathbf{G})$ follows from $\Phi \Vdash \mathbf{G}\mathsf{w}$ where w_α is a fresh parameter. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. Assume $\mathcal{M} \not\models \Pi^\alpha \mathbf{G}$. Then there must be some $\mathsf{a} \in \mathcal{D}_\alpha$ such that $\upsilon(\mathcal{E}(\mathbf{G})@\mathsf{a}) = \mathtt{F}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases) $$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w}\text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\,\mathfrak{NK}(\Pi)^{\mathsf{w}}$$

Suppose $\Phi \Vdash (\Pi^\alpha \mathbf{G})$ follows from $\Phi \Vdash \mathbf{G}\mathsf{w}$ where w_α is a fresh parameter. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. Assume $\mathcal{M} \not\models \Pi^\alpha \mathbf{G}$. Then there must be some $\mathsf{a} \in \mathcal{D}_\alpha$ such that $\upsilon(\mathcal{E}(\mathbf{G})@\mathsf{a}) = \mathtt{F}$. From $\mathcal{E}$, one can define $\mathcal{E}'$ such that $\mathcal{E}'(\mathsf{w}) = \mathsf{a}$ and $\mathcal{E}'_\varphi(\mathbf{A}_\alpha) = \mathcal{E}_\varphi(\mathbf{A}_\alpha)$ if w does not occur in $\mathbf{A}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases) $$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w}\text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\,\mathfrak{NK}(\Pi)^{\mathsf{w}}$$

Suppose $\Phi \Vdash (\Pi^\alpha \mathbf{G})$ follows from $\Phi \Vdash \mathbf{G}\mathsf{w}$ where w_α is a fresh parameter. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. Assume $\mathcal{M} \not\models \Pi^\alpha \mathbf{G}$. Then there must be some $\mathsf{a} \in \mathcal{D}_\alpha$ such that $\upsilon(\mathcal{E}(\mathbf{G})@\mathsf{a}) = \mathtt{F}$. From $\mathcal{E}$, one can define $\mathcal{E}'$ such that $\mathcal{E}'(\mathsf{w}) = \mathsf{a}$ and $\mathcal{E}'_\varphi(\mathbf{A}_\alpha) = \mathcal{E}_\varphi(\mathbf{A}_\alpha)$ if w does not occur in $\mathbf{A}$. Let $\mathcal{M}' := (\mathcal{D}, @, \mathcal{E}', \upsilon)$. One can check $\mathcal{M}' \in \mathfrak{M}_*(\Sigma)$ using the fact that $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases) $$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w}\text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\,\mathfrak{NK}(\Pi)^{\mathsf{w}}$$

Suppose $\Phi \Vdash (\Pi^\alpha \mathbf{G})$ follows from $\Phi \Vdash \mathbf{G}\mathsf{w}$ where w_α is a fresh parameter. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. Assume $\mathcal{M} \not\models \Pi^\alpha \mathbf{G}$. Then there must be some $\mathsf{a} \in \mathcal{D}_\alpha$ such that $\upsilon(\mathcal{E}(\mathbf{G})@\mathsf{a}) = \mathtt{F}$. From $\mathcal{E}$, one can define $\mathcal{E}'$ such that $\mathcal{E}'(\mathsf{w}) = \mathsf{a}$ and $\mathcal{E}'_\varphi(\mathbf{A}_\alpha) = \mathcal{E}_\varphi(\mathbf{A}_\alpha)$ if w does not occur in $\mathbf{A}$. Let $\mathcal{M}' := (\mathcal{D}, @, \mathcal{E}', \upsilon)$. One can check $\mathcal{M}' \in \mathfrak{M}_*(\Sigma)$ using the fact that $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$. Since $\mathcal{M}' \models \Phi$, by induction we have $\mathcal{M}' \models \mathbf{G}\mathsf{w}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \mathbf{G}\mathsf{w}_\alpha \quad \mathsf{w} \text{ new parameter}}{\Phi \Vdash \Pi^\alpha \mathbf{G}} \mathfrak{NK}(\Pi I)^{\mathsf{w}}$$

Suppose $\Phi \Vdash (\Pi^\alpha \mathbf{G})$ follows from $\Phi \Vdash \mathbf{G}\mathsf{w}$ where w_α is a fresh parameter. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. Assume $\mathcal{M} \not\models \Pi^\alpha \mathbf{G}$. Then there must be some $\mathsf{a} \in \mathcal{D}_\alpha$ such that $\upsilon(\mathcal{E}(\mathbf{G})@\mathsf{a}) = \mathtt{F}$. From $\mathcal{E}$, one can define $\mathcal{E}'$ such that $\mathcal{E}'(\mathsf{w}) = \mathsf{a}$ and $\mathcal{E}'_\varphi(\mathbf{A}_\alpha) = \mathcal{E}_\varphi(\mathbf{A}_\alpha)$ if w does not occur in $\mathbf{A}$. Let $\mathcal{M}' := (\mathcal{D}, @, \mathcal{E}', \upsilon)$. One can check $\mathcal{M}' \in \mathfrak{M}_*(\Sigma)$ using the fact that $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$. Since $\mathcal{M}' \models \Phi$, by induction we have $\mathcal{M}' \models \mathbf{G}\mathsf{w}$. This contradicts $\upsilon(\mathcal{E}'(\mathbf{G})@\mathsf{a}) = \upsilon(\mathcal{E}(\mathbf{G})@\mathsf{a}) = \mathtt{F}$. Thus, $\mathcal{M} \models \Pi^\alpha \mathbf{G}$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \Pi^\alpha \mathbf{G}}{\Phi \Vdash \mathbf{G}\mathbf{A}} \mathfrak{NK}(\Pi E)$$

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \Pi^\alpha \mathbf{G}}{\Phi \Vdash \mathbf{G}\mathbf{A}} \mathfrak{NK}(\Pi E)$$

Suppose $\mathbf{C}$ is $(\mathbf{G}\mathbf{A})$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\Pi^\alpha \mathbf{G})$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \Pi^\alpha \mathbf{G}}{\Phi \Vdash \mathbf{G}\mathbf{A}} \mathfrak{NK}(\Pi E)$$

Suppose $\mathbf{C}$ is $(\mathbf{G}\mathbf{A})$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\Pi^\alpha \mathbf{G})$. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \Pi^\alpha \mathbf{G}}{\Phi \Vdash \mathbf{GA}}\,\mathfrak{NK}(\Pi E)$$

Suppose $\mathbf{C}$ is $(\mathbf{GA})$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\Pi^\alpha \mathbf{G})$. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. By induction, $\mathcal{M} \models (\Pi^\alpha \mathbf{G})$ and thus $\upsilon(\mathcal{E}(\mathbf{G}))@\mathsf{a} = \mathtt{T}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \Pi^\alpha \mathbf{G}}{\Phi \Vdash \mathbf{GA}}\,\mathfrak{NK}(\Pi E)$$

Suppose $\mathbf{C}$ is $(\mathbf{GA})$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\Pi^\alpha \mathbf{G})$. Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. By induction, $\mathcal{M} \models (\Pi^\alpha \mathbf{G})$ and thus $\upsilon(\mathcal{E}(\mathbf{G}))@\mathsf{a} = \mathtt{T}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$. In particular, $\mathcal{M} \models \mathbf{GA}$.

q.e.d.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\eta)$$

(In this case $*$ contains property η)
Analogous to $\mathfrak{NK}(\beta)$ using property η

q.e.d.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \forall \mathsf{x}_\alpha. \mathbf{M} \doteq^\beta \mathbf{N}}{\Phi \Vdash (\lambda \mathsf{x}_\alpha. \mathbf{M}) \doteq^{\beta\alpha} (\lambda \mathsf{x}_\alpha. \mathbf{N})}\,\mathfrak{NK}(\xi)$$

(In this case $*$ contains property ξ)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. By induction, we have $\mathcal{M} \models \forall \mathsf{X}_\alpha. \mathbf{M} \doteq^\beta \mathbf{N}$. So, for any assignment φ and $\mathsf{a} \in \mathcal{D}_\alpha$, $\mathcal{M} \models_{\varphi,[\mathsf{a}/\mathsf{X}]} \mathbf{M} \doteq^\beta \mathbf{N}$. Since property $\mathfrak{q}$ holds, by a previous Lemma we have $\mathcal{E}_{\varphi,[\mathsf{a}/\mathsf{X}]}(\mathbf{M}) = \mathcal{E}_{\varphi,[\mathsf{a}/\mathsf{X}]}(\mathbf{N})$. By property ξ, $\mathcal{E}_\varphi(\lambda \mathsf{X}_\alpha. \mathbf{M}) = \mathcal{E}_\varphi(\lambda \mathsf{X}_\alpha. \mathbf{N})$ and thus $\mathcal{M} \models \mathbf{C}$ by a previous Lemma.

q.e.d.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$.
That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi \Vdash \forall X_\alpha . \mathbf{G}X \doteq^\beta \mathbf{H}X}{\Phi \Vdash \mathbf{G} \doteq^{\beta\alpha} \mathbf{H}}\,\mathfrak{NK}(\mathfrak{f})$$

(In this case $*$ contains property $\mathfrak{f}$)
Let $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. By induction, we know $\mathcal{M} \models \forall X_\alpha . \mathbf{G}X \doteq^\beta \mathbf{H}X$. Note that property $\mathfrak{q}$ holds for $\mathcal{M}$ since $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$. By a previous theorem, we must have $\mathcal{M} \models (\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H})$.

q.e.d.

Soundness of $\mathfrak{NK}_*$

Thm.: $\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*(\Sigma)$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$.
That is, if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$ is derivable, then $\mathcal{M} \models \mathbf{C}$ for all models $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*(\Sigma)$ such that $\mathcal{M} \models \Phi$.

Proof: (step cases)

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{B} \quad \Phi * \mathbf{B} \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \doteq^{o} \mathbf{B}}\,\mathfrak{NK}(\mathfrak{b})$$

(In this case $*$ contains property $\mathfrak{b}$)
Let $\mathcal{M} = (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*(\Sigma)$ be a model of Φ. If $\mathcal{M} \models \mathbf{A}$, then $\mathcal{M} \models \mathbf{B}$ by induction. If $\mathcal{M} \models \mathbf{B}$, then $\mathcal{M} \models \mathbf{A}$ by induction. These facts imply $\upsilon(\mathcal{E}(\mathbf{A})) = \upsilon(\mathcal{E}(\mathbf{B}))$. By a previous lemma, we have $\mathcal{M} \models (\mathbf{A} \Leftrightarrow \mathbf{B})$. By a previous theorem, we must have $\mathcal{M} \models (\mathbf{A} \doteq^{o} \mathbf{B})$.

q.e.d.

Completeness of $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Completeness of $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Completeness of $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Proof:

Completeness of $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Proof:

How can we easily prove this?

Completeness (of $\mathfrak{NK}_*$)

- Completeness can be proven rather easily for propositional logic calculi.

Completeness (of $\mathfrak{NK}_*$)

- Completeness can be proven rather easily for propositional logic calculi.
- For first-order and especially higher-order logic completeness proofs become increasingly difficult and technical.

Completeness (of $\mathfrak{NK}_*$)

- Completeness can be proven rather easily for propositional logic calculi.
- For first-order and especially higher-order logic completeness proofs become increasingly difficult and technical.
- Here we will introduce a strong proof tool that uniformly supports completeness proofs (and many other things): abstract consistency.

Completeness (of $\mathfrak{NK}_*$)

- Completeness can be proven rather easily for propositional logic calculi.
- For first-order and especially higher-order logic completeness proofs become increasingly difficult and technical.
- Here we will introduce a strong proof tool that uniformly supports completeness proofs (and many other things): abstract consistency.
- This proof tool is based on a strong theorem which connects syntax and semantics: model existence theorem.

Completeness (of $\mathfrak{NK}_*$)

- Completeness can be proven rather easily for propositional logic calculi.
- For first-order and especially higher-order logic completeness proofs become increasingly difficult and technical.
- Here we will introduce a strong proof tool that uniformly supports completeness proofs (and many other things): abstract consistency.
- This proof tool is based on a strong theorem which connects syntax and semantics: model existence theorem.

HOL Test Problems

Recommendation:

- if you develop provers/calculi for simple type theory then

HOL Test Problems

Recommendation:

- if you develop provers/calculi for simple type theory then
- first empirically analyse soundness and completeness wrt to our model classes

HOL Test Problems

Recommendation:

- if you develop provers/calculi for simple type theory then
- first empirically analyse soundness and completeness wrt to our model classes
- with the help of examples (published in [TPHOLs-05])

HOL Test Problems

Recommendation:

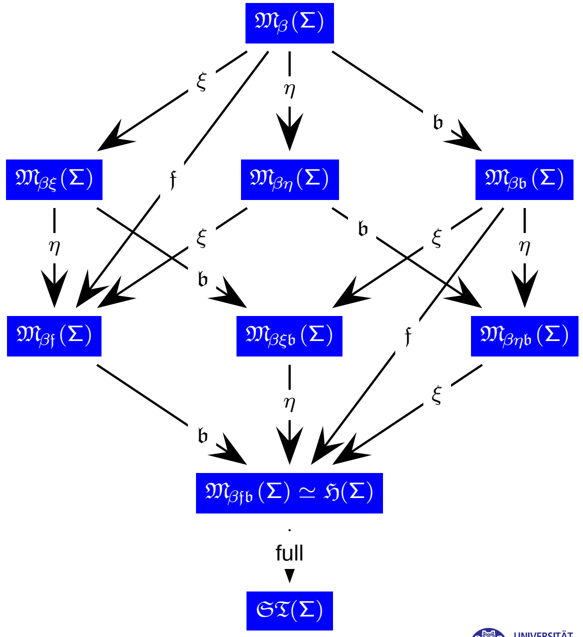
- if you develop provers/calculi for simple type theory then
- first empirically analyse soundness and completeness wrt to our model classes
- with the help of examples (published in [TPHOLs-05])
- before you formally analyse them

HOL Test Problems

Recommendation:

- if you develop provers/calculi for simple type theory then
- first empirically analyse soundness and completeness wrt to our model classes
- with the help of examples (published in [TPHOLs-05])
- before you formally analyse them
- with the help of the abstract consistency proof method (published in [JSL-04] and [Unpublished-04])

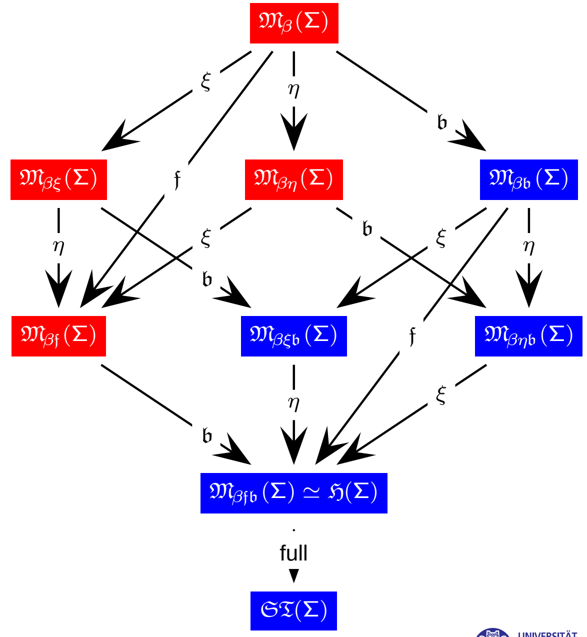
HOL Test Problems (from before)



- $\forall X.\forall y.X \lor y \Leftrightarrow y \lor X$

valid for all model classes

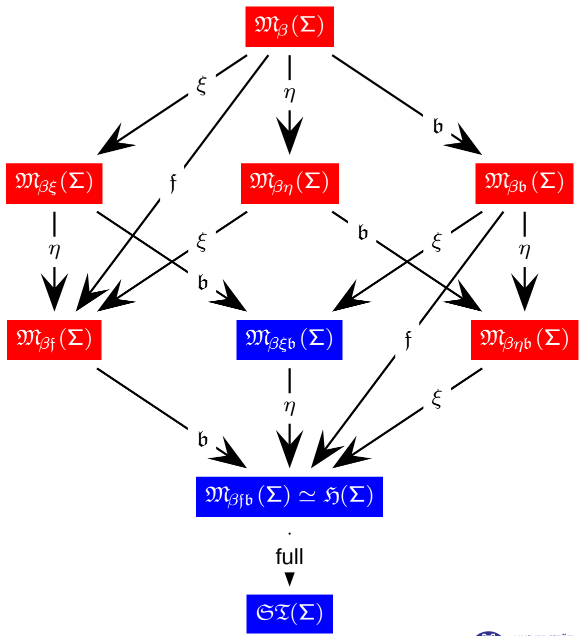
HOL Test Problems (from before)



- $\forall X.\forall Y.X \lor Y \Leftrightarrow Y \lor X$
- $\forall X.\forall Y.X \lor Y \doteq Y \lor X$

validity requires $\mathfrak{b}$

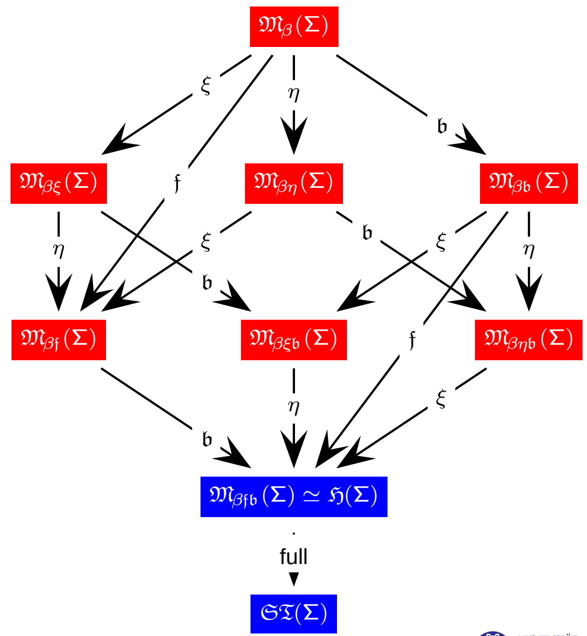
HOL Test Problems (from before)



- $\forall X.\forall Y.X \lor Y \Leftrightarrow Y \lor X$
- $\forall X.\forall Y.X \lor Y \doteq Y \lor X$
- $\lambda X.\lambda Y.X \lor Y \doteq \lambda X.\lambda Y.Y \lor X$

validity requires $\mathfrak{b}$ and ξ

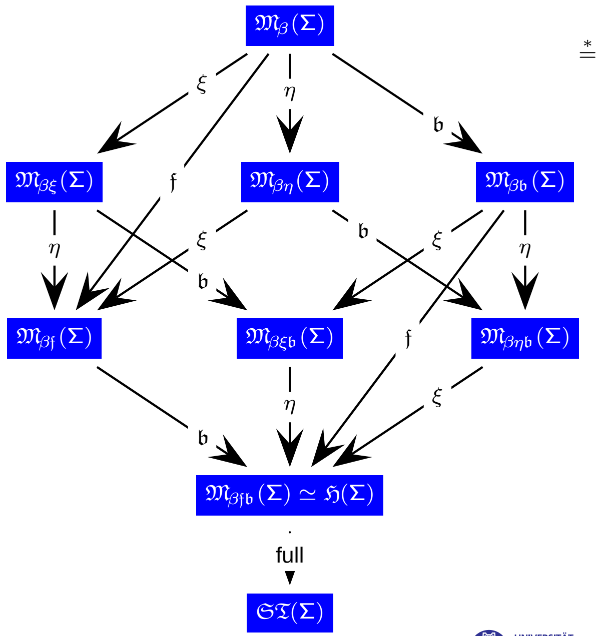
HOL Test Problems (from before)



- $\forall X.\forall Y.X \lor Y \Leftrightarrow Y \lor X$
- $\forall X.\forall Y.X \lor Y \doteq Y \lor X$
- $\lambda X.\lambda Y.X \lor Y \doteq \lambda X.\lambda Y.Y \lor X$
- $\lor \doteq \lambda X.\lambda Y.Y \lor X$

validity requires $\mathfrak{b}$ and $\mathfrak{f}$

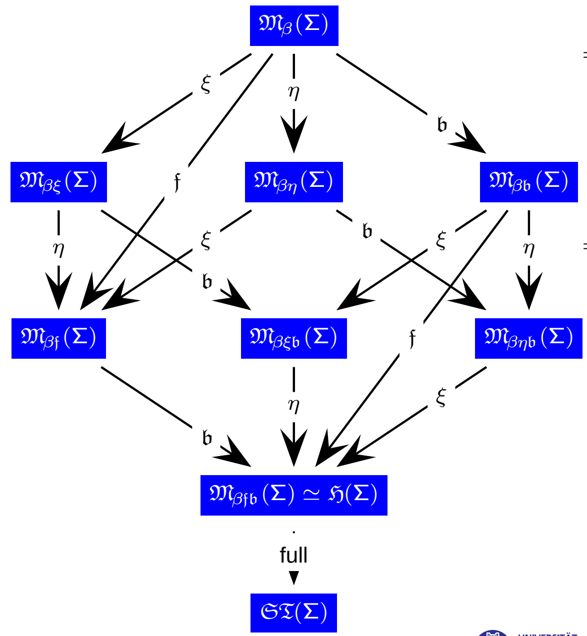
Other HOL Test Problems: β



$\stackrel{*}{=}$ **is equivalence relation**

- $\forall X_\alpha . X \stackrel{*}{=} X$
- $\forall X_\alpha, Y_\alpha . X \stackrel{*}{=} Y \supset Y \stackrel{*}{=} X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha . (X \stackrel{*}{=} Y \wedge Y \stackrel{*}{=} Z) \supset X \stackrel{*}{=} Z$

Other HOL Test Problems: β



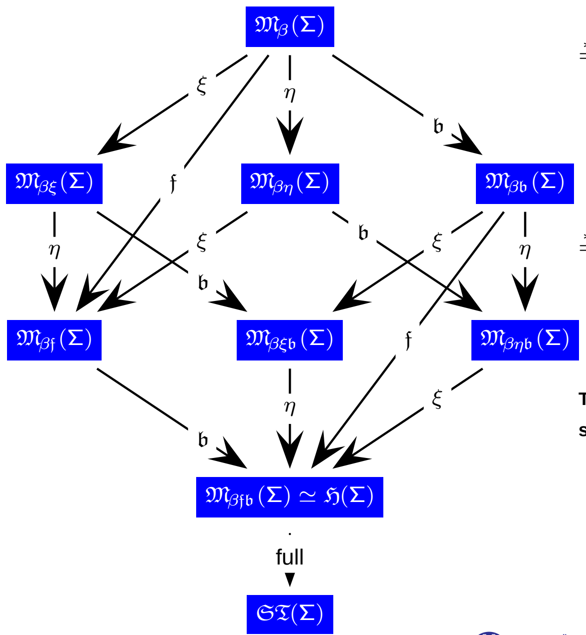
$\stackrel{*}{=}$ **is equivalence relation**

- $\forall X_\alpha . X \stackrel{*}{=} X$
- $\forall X_\alpha, Y_\alpha . X \stackrel{*}{=} Y \supset Y \stackrel{*}{=} X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha . (X \stackrel{*}{=} Y \wedge Y \stackrel{*}{=} Z) \supset X \stackrel{*}{=} Z$

$\stackrel{*}{=}$ **is congruence relation**

- $\forall X_\alpha, Y_\alpha, F_{\alpha\alpha} . X \stackrel{*}{=} Y \supset (FX) \stackrel{*}{=} (FY)$
- $\forall X_\alpha, Y_\alpha, P_{o\alpha} . X \stackrel{*}{=} Y \wedge (PX) \supset (PY)$

Other HOL Test Problems: β



$\stackrel{*}{=}$ **is equivalence relation**

- $\forall X_\alpha . X \stackrel{*}{=} X$
- $\forall X_\alpha, Y_\alpha . X \stackrel{*}{=} Y \supset Y \stackrel{*}{=} X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha . (X \stackrel{*}{=} Y \wedge Y \stackrel{*}{=} Z) \supset X \stackrel{*}{=} Z$

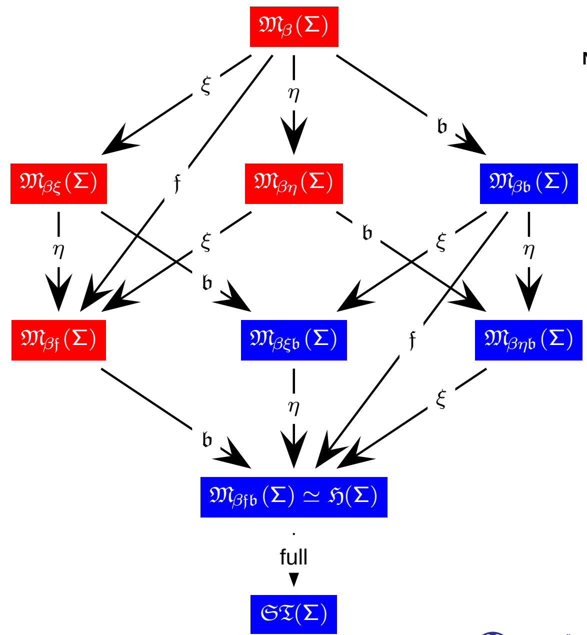
$\stackrel{*}{=}$ **is congruence relation**

- $\forall X_\alpha, Y_\alpha, F_{\alpha\alpha} . X \stackrel{*}{=} Y \supset (FX) \stackrel{*}{=} (FY)$
- $\forall X_\alpha, Y_\alpha, P_{o\alpha} . X \stackrel{*}{=} Y \wedge (PX) \supset (PY)$

Trivial directions of Boolean and functional extensionality

- $\forall A_o, B_o . A \stackrel{*}{=} B \supset (A \Leftrightarrow B)$
- $\forall F_{\beta\alpha}, G_{\beta\alpha} . F \stackrel{*}{=} G \supset (\forall X_\alpha . FX \stackrel{*}{=} GX)$

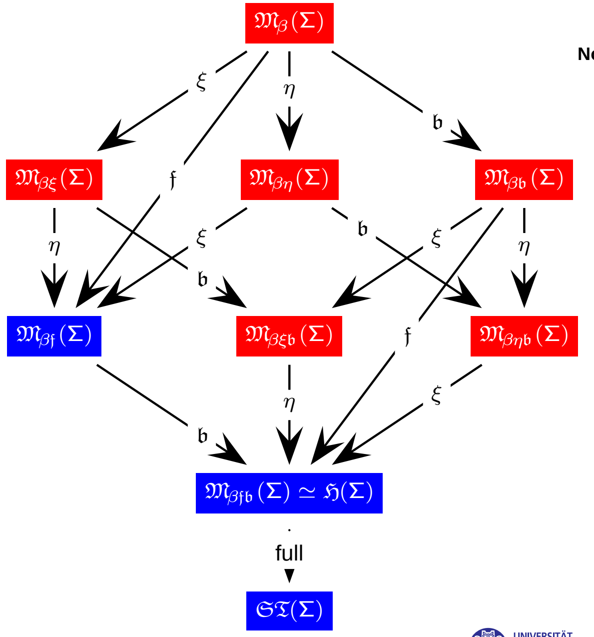
Other HOL Test Problems: $\mathfrak{b}$



Non-trivial direction of Boolean extensionality

- $\forall A_o, B_o . (A \Leftrightarrow B) \supset A \stackrel{*}{=} B$

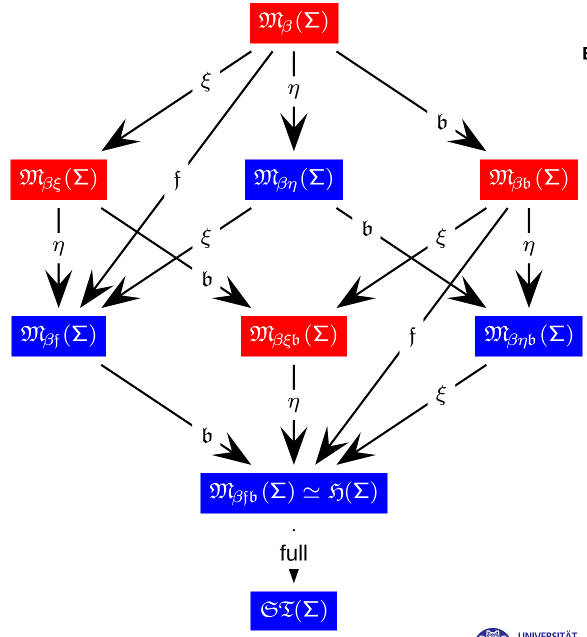
Other HOL Test Problems: $\mathfrak{f}$



Non-trivial direct. of functional extensionality

- $\forall F_{\beta\alpha}, G_{\beta\alpha}.(\forall X_\alpha.FX \stackrel{*}{\doteq} GX) \supset F \stackrel{*}{\doteq} G$

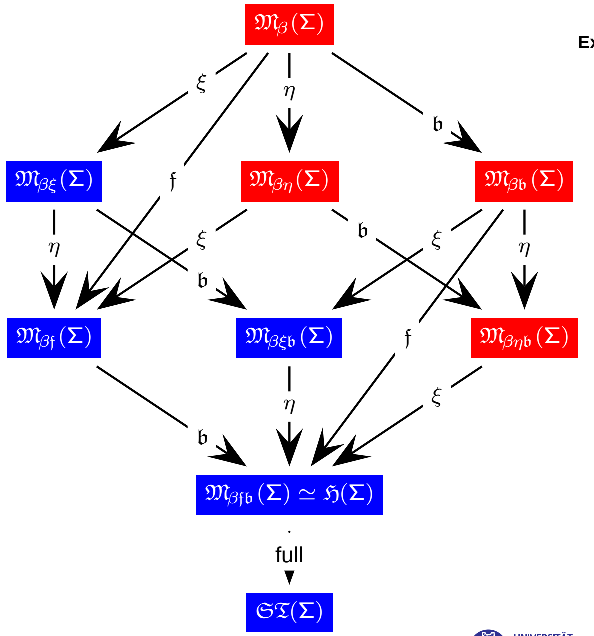
Other HOL Test Problems: η



Example requiring property η

- $(p_{o(\iota\iota)}(\lambda X_\iota.f_{\iota\iota}X)) \supset (p\ f)$

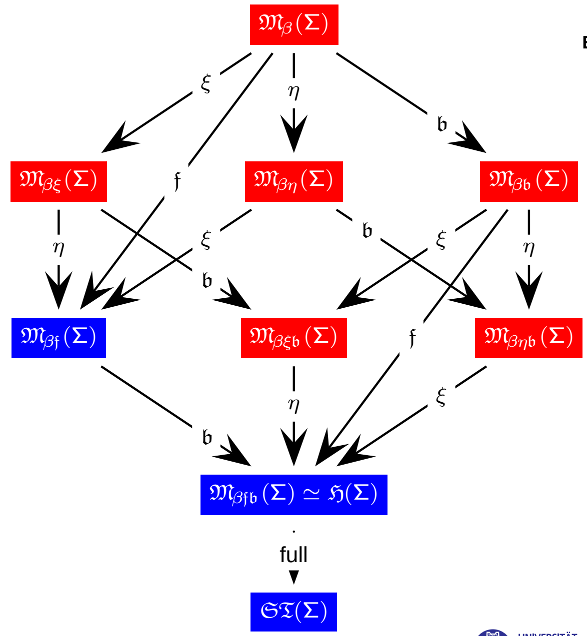
Other HOL Test Problems: ξ



Example requiring property ξ (and $\mathfrak{q}$!)

- $(\forall X_\iota.(f_{\iota\iota}X) \stackrel{*}{\doteq} X) \wedge p_{o(\iota\iota)}(\lambda X_\iota.X)$
  $\supset p(\lambda X_\iota.fX)$

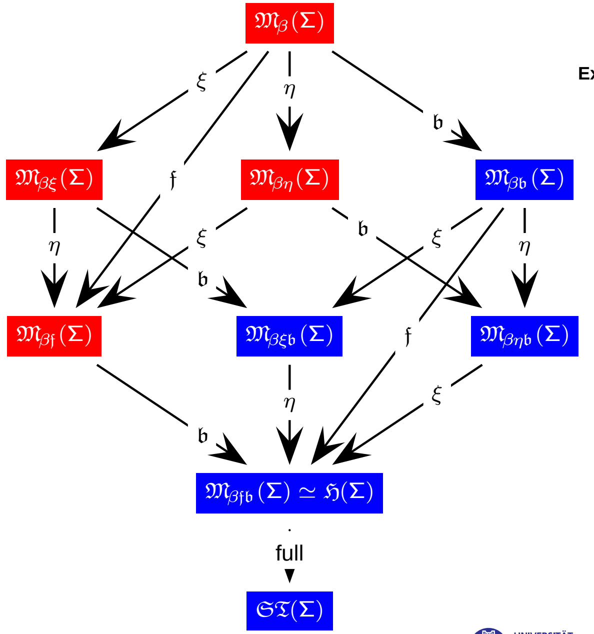
Other HOL Test Problems: $\mathfrak{f}$



Example requiring property $\mathfrak{f}$ (and $\mathfrak{q}$!)

- $(\forall X_\iota.(f_{\iota\iota}X) \stackrel{*}{\doteq} X) \wedge p_{o(\iota\iota)}(\lambda X_\iota.X)$
  $\supset (p\ f)$

Other HOL Test Problems: $\mathfrak{b}$



Examples requiring property $\mathfrak{b}$

- $(p_{oo}\ a_o) \wedge (p\ b_o) \Rightarrow (p\ (a \wedge b))$
- $\neg(a \stackrel{*}{\doteq} \neg a)$ (in particular $\neg(a = \neg a)$)
- $(h_{\iota o}((h\top) \stackrel{*}{\doteq} (h\bot))) \stackrel{*}{\doteq} (h\bot)$

Abstract Consistency

Abstract Consistency: History

- Technique was developed for first-order logic by Jaakko Hintikka and Raymond Smullyan [Hintikka55,Smullyan63,Smullyan68]. It is well explained in Fitting's textbook [Fitting96].

Abstract Consistency: History

- Technique was developed for first-order logic by Jaakko Hintikka and Raymond Smullyan [Hintikka55,Smullyan63,Smullyan68]. It is well explained in Fitting's textbook [Fitting96].

- Model Existence Theorem before presented by Henkin.

Abstract Consistency: History

- Technique was developed for first-order logic by Jaakko Hintikka and Raymond Smullyan [Hintikka55,Smullyan63,Smullyan68]. It is well explained in Fitting's textbook [Fitting96].
- Model Existence Theorem before presented by Henkin.
- The technique has been (partly) extended to higher-order logic by Peter Andrews' in [Andrews71]; he only achieves a generalization for a rather weak notion od semantics (corresponding to our $\mathfrak{M}_\beta(\Sigma)$).

Abstract Consistency: History

- Technique was developed for first-order logic by Jaakko Hintikka and Raymond Smullyan [Hintikka55,Smullyan63,Smullyan68]. It is well explained in Fitting's textbook [Fitting96].
- Model Existence Theorem before presented by Henkin.
- The technique has been (partly) extended to higher-order logic by Peter Andrews' in [Andrews71]; he only achieves a generalization for a rather weak notion od semantics (corresponding to our $\mathfrak{M}_\beta(\Sigma)$).
- The technique has been extended to our landscape of HOL model classes in [Chris-PhD-99,Chad-PhD-04,JSL-04].

Abstract Consistency: Idea

- A model existence theorem for a logical system (i.e., a logical language L together with a consequence relation $\models$) is a theorem of the form:

Abstract Consistency: Idea

- A model existence theorem for a logical system (i.e., a logical language L together with a consequence relation $\models$) is a theorem of the form:

  If a set of sentences Φ of L is a member of an (saturated) abstract consistency class Γ, then there exists a model for Φ.

Abstract Consistency: Idea

- Employing the model existence theorem we can prove completeness of a calculus $\mathcal{C}$ (i.e., the derivability rel. $\vdash_{\mathcal{C}}$) by

Abstract Consistency: Idea

- Employing the model existence theorem we can prove completeness of a calculus $\mathcal{C}$ (i.e., the derivability rel. $\vdash_{\mathcal{C}}$) by

  proving that the class Γ of sets of sentences Φ that are $\mathcal{C}$-consistent (i.e., cannot be refuted in $\mathcal{C}$: $\{\Phi|\Phi \nvdash_{\mathcal{C}} \mathbf{F}_o\}$) is an (saturated) abstract consistency class.

© Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[4] – p.183

Abstract Consistency: Idea

- Employing the model existence theorem we can prove completeness of a calculus $\mathcal{C}$ (i.e., the derivability rel. $\vdash_{\mathcal{C}}$) by

  proving that the class Γ of sets of sentences Φ that are $\mathcal{C}$-consistent (i.e., cannot be refuted in $\mathcal{C}$: $\{\Phi|\Phi \nvdash_{\mathcal{C}} \mathbf{F}_o\}$) is an (saturated) abstract consistency class.

- Why does this work?

© Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[4] – p.183

Abstract Consistency: Idea

- Employing the model existence theorem we can prove completeness of a calculus $\mathcal{C}$ (i.e., the derivability rel. $\vdash_{\mathcal{C}}$) by

  proving that the class Γ of sets of sentences Φ that are $\mathcal{C}$-consistent (i.e., cannot be refuted in $\mathcal{C}$: $\{\Phi|\Phi \nvdash_{\mathcal{C}} \mathbf{F}_o\}$) is an (saturated) abstract consistency class.

- Why does this work?
  - The model existence theorem tells us that $\mathcal{C}$-consistent sets of sentences are satisfiable.

© Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[4] – p.183

Abstract Consistency: Idea

- Employing the model existence theorem we can prove completeness of a calculus C (i.e., the derivability rel. $\vdash_C$) by

  proving that the class Γ of sets of sentences Φ that are C-consistent (i.e., cannot be refuted in C: $\{\Phi | \Phi \not\vdash_C \mathbf{F}_o\}$) is an (saturated) abstract consistency class.

- Why does this work?
  - The model existence theorem tells us that C-consistent sets of sentences are satisfiable.
  - Now we assume that a sentence A is valid, so $\neg A$ does not have a model and is therefore C-inconsistent.

Abstract Consistency: Idea

- Employing the model existence theorem we can prove completeness of a calculus C (i.e., the derivability rel. $\vdash_C$) by

  proving that the class Γ of sets of sentences Φ that are C-consistent (i.e., cannot be refuted in C: $\{\Phi | \Phi \not\vdash_C \mathbf{F}_o\}$) is an (saturated) abstract consistency class.

- Why does this work?
  - The model existence theorem tells us that C-consistent sets of sentences are satisfiable.
  - Now we assume that a sentence A is valid, so $\neg A$ does not have a model and is therefore C-inconsistent.
  - Hence, $\neg A$ is refutable in C.

©Benzmüller & Brown, 2006
UNIVERSITÄT DES SAARLANDES
SEMHOL'06-[4] – p.183

Abstract Consistency: Idea

- Employing the model existence theorem we can prove completeness of a calculus C (i.e., the derivability rel. $\vdash_C$) by

  proving that the class Γ of sets of sentences Φ that are C-consistent (i.e., cannot be refuted in C: $\{\Phi | \Phi \not\vdash_C \mathbf{F}_o\}$) is an (saturated) abstract consistency class.

- Why does this work?
  - The model existence theorem tells us that C-consistent sets of sentences are satisfiable.
  - Now we assume that a sentence A is valid, so $\neg A$ does not have a model and is therefore C-inconsistent.
  - Hence, $\neg A$ is refutable in C.
  - This shows refutation completeness of C.

©Benzmüller & Brown, 2006
UNIVERSITÄT DES SAARLANDES
SEMHOL'06-[4] – p.183

Abstract Consistency: Idea

- Employing the model existence theorem we can prove completeness of a calculus C (i.e., the derivability rel. $\vdash_C$) by

  proving that the class Γ of sets of sentences Φ that are C-consistent (i.e., cannot be refuted in C: $\{\Phi | \Phi \not\vdash_C \mathbf{F}_o\}$) is an (saturated) abstract consistency class.

- Why does this work?
  - The model existence theorem tells us that C-consistent sets of sentences are satisfiable.
  - Now we assume that a sentence A is valid, so $\neg A$ does not have a model and is therefore C-inconsistent.
  - Hence, $\neg A$ is refutable in C.
  - This shows refutation completeness of C.
  - For many calculi C, this also shows A is provable, thus establishing completeness of C.

©Benzmüller & Brown, 2006
UNIVERSITÄT DES SAARLANDES
SEMHOL'06-[4] – p.183

Def.: Closed under Subsets / Compact

Defn.: Let C be a class of sets then C is called closed under subset if for all sets S and T it holds that

Def.: Closed under Subsets / Compact

Defn.: Let C be a class of sets then C is called closed under subset if for all sets S and T it holds that

from $S \subseteq T$ and $T \in C$ it follows that $S \in C$.

Def.: Closed under Subsets / Compact

Defn.: Let C be a class of sets then C is called closed under subset if for all sets S and T it holds that

from $S \subseteq T$ and $T \in C$ it follows that $S \in C$.

Defn.: Let C be a class of sets. C is called compact or of finite character if and only if for every set S holds:

Def.: Closed under Subsets / Compact

Defn.: Let C be a class of sets then C is called closed under subset if for all sets S and T it holds that

from $S \subseteq T$ and $T \in C$ it follows that $S \in C$.

Defn.: Let C be a class of sets. C is called compact or of finite character if and only if for every set S holds:

$S \in C$ if and only if every finite subset of S is a member of C.

Ex.: Closed under Subsets / Compact

- not closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg A\}\}$

Ex.: Closed under Subsets / Compact

- not closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg A\}\}$
- closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B), C\}, \{\neg A, C\}, \{\neg(A \vee B)\}, \{\neg A\}, \{C\}, \{\}\}$

Ex.: Closed under Subsets / Compact

- not closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg A\}\}$
- closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B), C\}, \{\neg A, C\}, \{\neg(A \vee B)\}, \{\neg A\}, \{C\}, \{\}\}$
- We define two classes of sets

Ex.: Closed under Subsets / Compact

- not closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg A\}\}$
- closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B), C\}, \{\neg A, C\}, \{\neg(A \vee B)\}, \{\neg A\}, \{C\}, \{\}\}$
- We define two classes of sets
  - $C := \{\varphi \mid \varphi$ is finite subset of $\mathbb{N}\}$

Ex.: Closed under Subsets / Compact

- not closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg A\}\}$
- closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B), C\}, \{\neg A, C\}, \{\neg(A \vee B)\}, \{\neg A\}, \{C\}, \{\}\}$
- We define two classes of sets
  - $C := \{\varphi \mid \varphi$ is finite subset of $\mathbb{N}\}$
  - $D := 2^{\mathbb{N}}$

Ex.: Closed under Subsets / Compact

- not closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg A\}\}$
- closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B), C\}, \{\neg A, C\}, \{\neg(A \vee B)\}, \{\neg A\}, \{C\}, \{\}\}$
- We define two classes of sets
  - $C := \{\varphi \mid \varphi$ is finite subset of $\mathbb{N}\}$
  - $D := 2^{\mathbb{N}}$
  - C is closed under subsets but not compact.

Ex.: Closed under Subsets / Compact

- not closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg A\}\}$
- closed under subsets: $\{\{\neg(A \vee B), \neg A, C\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B), C\}, \{\neg A, C\}, \{\neg(A \vee B)\}, \{\neg A\}, \{C\}, \{\}\}$
- We define two classes of sets
  - $C := \{\varphi \mid \varphi$ is finite subset of $\mathbb{N}\}$
  - $D := 2^{\mathbb{N}}$
  - C is closed under subsets but not compact.
  - D is closed under subsets and compact.

Closed under Subsets / Compact

Lemma:

If C is compact then C is closed under subsets.

Closed under Subsets / Compact

Lemma:

If C is compact then C is closed under subsets.

Proof:

Closed under Subsets / Compact

Lemma:

If C is compact then C is closed under subsets.

Proof:

Let $T \in C$ and $S \subseteq T$.

Closed under Subsets / Compact

Lemma:

If C is compact then C is closed under subsets.

Proof:

Let $T \in C$ and $S \subseteq T$.

We have to show that $S \in C$.

Closed under Subsets / Compact

Lemma:

If C is compact then C is closed under subsets.

Proof:

Let $T \in C$ and $S \subseteq T$.
We have to show that $S \in C$.

Every finite subset A of S is also a finite subset of T.

Closed under Subsets / Compact

Lemma:

If C is compact then C is closed under subsets.

Proof:

Let $T \in C$ and $S \subseteq T$.
We have to show that $S \in C$.
Every finite subset A of S is also a finite subset of T.
Since C is compact and $T \in C$ we get that all $A \in C$.

Closed under Subsets / Compact

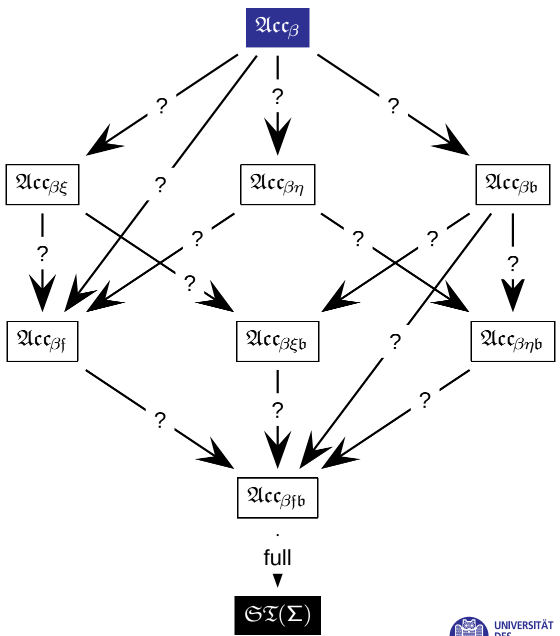
Lemma:

If C is compact then C is closed under subsets.

Proof:

Let $T \in C$ and $S \subseteq T$.
We have to show that $S \in C$.
Every finite subset A of S is also a finite subset of T.
Since C is compact and $T \in C$ we get that all $A \in C$.
Thus, $S \in C$ by compactness.

Basic Abstract Consistency Properties



Basic Abstract Consistency Properties

Defn.: Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha \to o}(\Sigma)$ are arbitrary):

Basic Abstract Consistency Properties

Defn.: Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$ are arbitrary):

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

Basic Abstract Consistency Properties

Defn.: Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$ are arbitrary):

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

Basic Abstract Consistency Properties

Defn.: Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$ are arbitrary):

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

Basic Abstract Consistency Properties

Defn.: Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$ are arbitrary):

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

Basic Abstract Consistency Properties

Defn.: Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$ are arbitrary):

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$.

Basic Abstract Consistency Properties

Defn.: Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$ are arbitrary):

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\forall$ If $\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \mathbf{F}\mathbf{W} \in \Gamma_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

Basic Abstract Consistency Properties

Defn.: Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$ are arbitrary):

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\forall$ If $\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \mathbf{F}\mathbf{W} \in \Gamma_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

$\nabla_\exists$ If $\neg\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \neg(\mathbf{F}w) \in \Gamma_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Basic Abstract Consistency Properties

Defn.: Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$ are arbitrary):

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\forall$ If $\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \mathbf{F}\mathbf{W} \in \Gamma_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

$\nabla_\exists$ If $\neg\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \neg(\mathbf{F}w) \in \Gamma_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

(These properties are going back to Hintikka, Smullyan, and Andrews)

Abstract Consistency Class $\mathfrak{Acc}_\beta$

Defn.: Let Σ be a signature and Γ_Σ be a class of sets of Σ-sentences that is closed under subsets.

Abstract Consistency Class $\mathfrak{Acc}_\beta$

Defn.: Let Σ be a signature and Γ_Σ be a class of sets of Σ-sentences that is closed under subsets.

- If $\nabla_c, \nabla_\neg, \nabla_\beta, \nabla_\vee, \nabla_\wedge, \nabla_\forall$ and $\nabla_\exists$ are valid for Γ_Σ, then Γ_Σ is called an abstract consistency class for Σ-models.

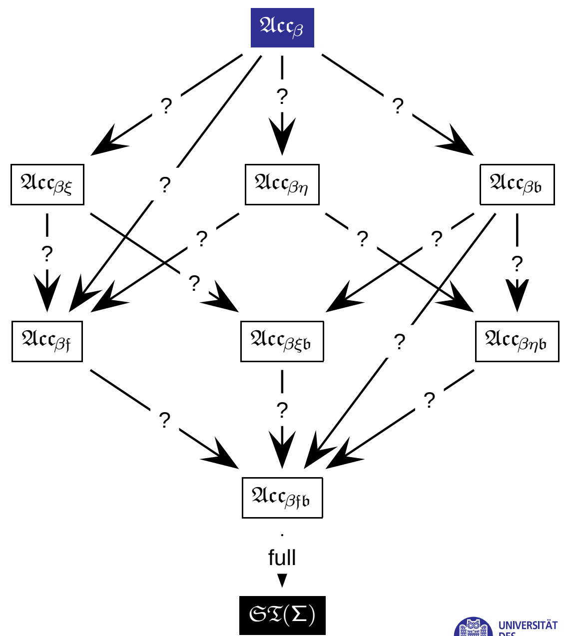
Abstract Consistency Class $\mathfrak{Acc}_\beta$

Defn.: Let Σ be a signature and Γ_Σ be a class of sets of Σ-sentences that is closed under subsets.

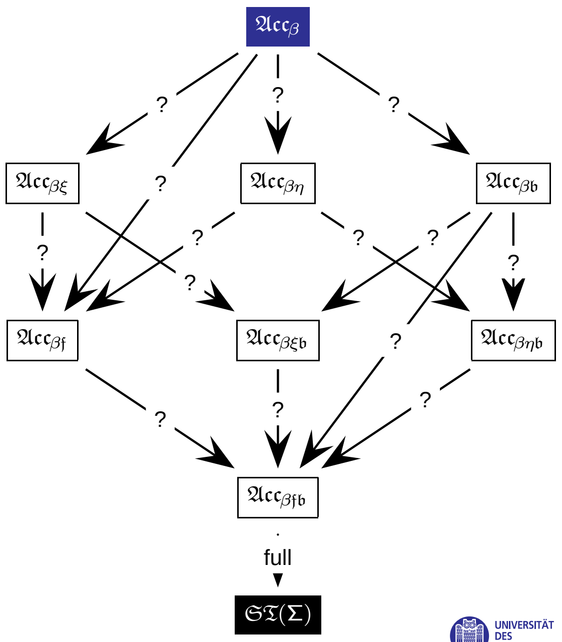
- If $\nabla_c, \nabla_\neg, \nabla_\beta, \nabla_\vee, \nabla_\wedge, \nabla_\forall$ and $\nabla_\exists$ are valid for Γ_Σ, then Γ_Σ is called an abstract consistency class for Σ-models.

- We will denote the collection of abstract consistency classes by $\mathfrak{Acc}_\beta$.

Basic Abstract Consistency Properties



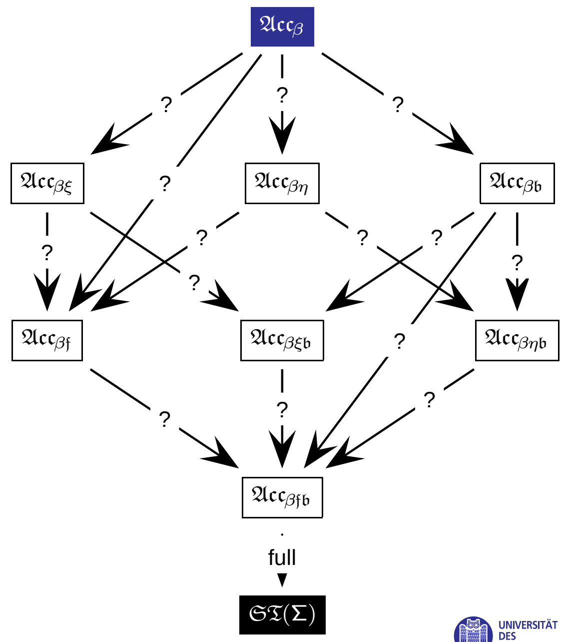
Basic Abstract Consistency Properties



Properties for $\mathfrak{Acc}_\beta$

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

Basic Abstract Consistency Properties

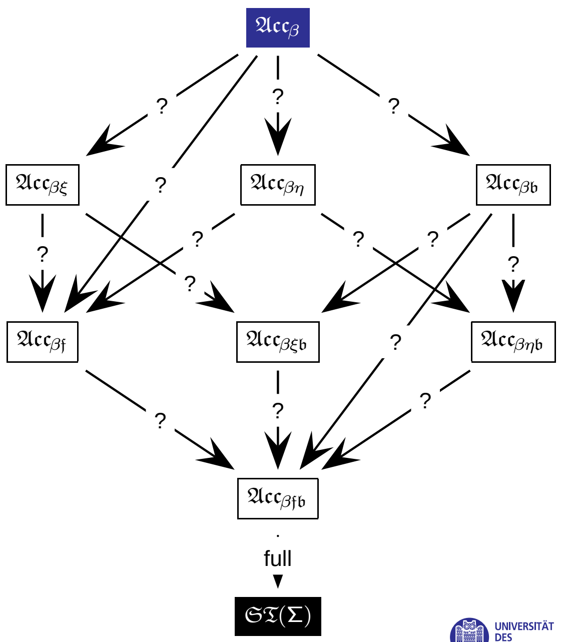


Properties for $\mathfrak{Acc}_\beta$

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

Basic Abstract Consistency Properties



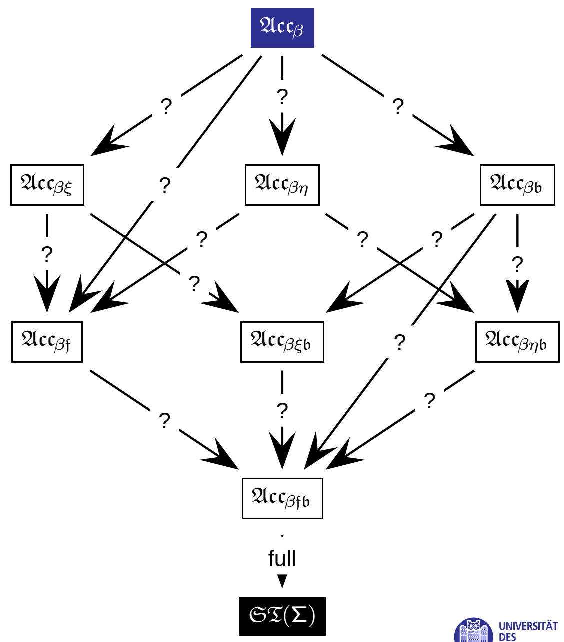
Properties for $\mathfrak{Acc}_\beta$

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

Basic Abstract Consistency Properties



Properties for $\mathfrak{Acc}_\beta$

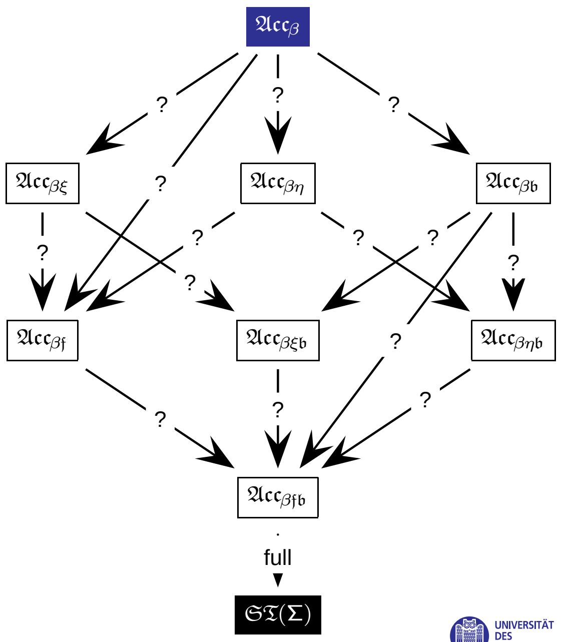
∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

$\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

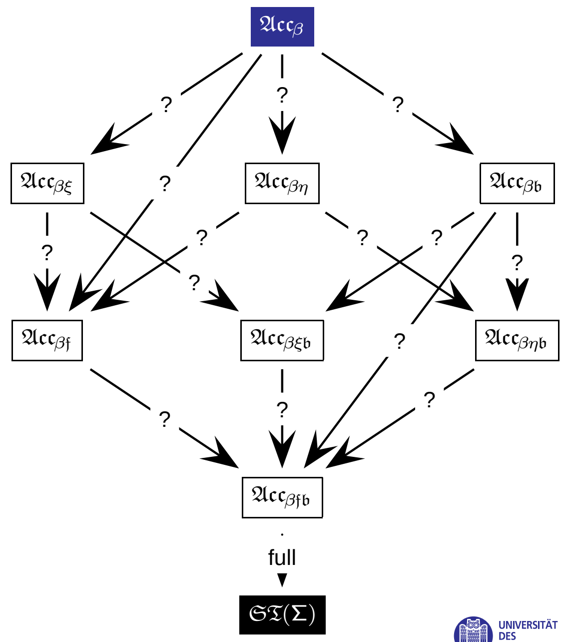
Basic Abstract Consistency Properties



Properties for $\mathfrak{Acc}_\beta$

- ∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.
- $\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.
- ∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.
- $\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma$.
- $\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$.

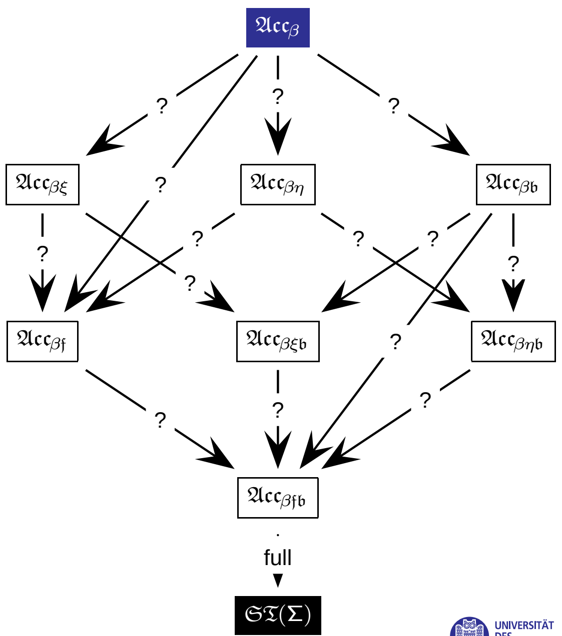
Basic Abstract Consistency Properties



Properties for $\mathfrak{Acc}_\beta$

- ∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.
- $\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.
- ∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.
- $\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma$.
- $\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$.
- $\nabla_\forall$ If $\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \mathbf{F}\mathbf{W} \in \Gamma_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

Basic Abstract Consistency Properties



Properties for $\mathfrak{Acc}_\beta$

- ∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.
- $\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$.
- ∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.
- $\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma$.
- $\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$.
- $\nabla_\forall$ If $\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \mathbf{F}\mathbf{W} \in \Gamma_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.
- $\nabla_\exists$ If $\neg\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \neg(\mathbf{F}w) \in \Gamma_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Extens. Abstract Consistency Properties

Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{G}, \mathbf{H}, (\lambda X_\alpha.\mathbf{M}), (\lambda X_\alpha.\mathbf{N}) \in cwff_{\alpha\to\beta}(\Sigma)$ are arbitrary):

Extens. Abstract Consistency Properties

Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{G}, \mathbf{H}, (\lambda X_\alpha . \mathbf{M}), (\lambda X_\alpha . \mathbf{N}) \in \mathit{cwff}_{\alpha \to \beta}(\Sigma)$ are arbitrary):

$\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^{o} \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$.

Extens. Abstract Consistency Properties

Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{G}, \mathbf{H}, (\lambda X_\alpha . \mathbf{M}), (\lambda X_\alpha . \mathbf{N}) \in \mathit{cwff}_{\alpha \to \beta}(\Sigma)$ are arbitrary):

$\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^{o} \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$.

∇_η If $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[4] – p.191

Extens. Abstract Consistency Properties

Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{G}, \mathbf{H}, (\lambda X_\alpha . \mathbf{M}), (\lambda X_\alpha . \mathbf{N}) \in \mathit{cwff}_{\alpha \to \beta}(\Sigma)$ are arbitrary):

$\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^{o} \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$.

∇_η If $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

∇_ξ If $\neg(\lambda X_\alpha . \mathbf{M} \doteq^{\alpha \to \beta} \lambda X_\alpha . \mathbf{N}) \in \Phi$, then $\Phi * \neg([w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N}) \in \Gamma_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[4] – p.191

Extens. Abstract Consistency Properties

Let Γ_Σ be a class of sets of Σ-sentences. We define (where $\Phi \in \Gamma_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{G}, \mathbf{H}, (\lambda X_\alpha . \mathbf{M}), (\lambda X_\alpha . \mathbf{N}) \in \mathit{cwff}_{\alpha \to \beta}(\Sigma)$ are arbitrary):

$\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^{o} \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$.

∇_η If $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

∇_ξ If $\neg(\lambda X_\alpha . \mathbf{M} \doteq^{\alpha \to \beta} \lambda X_\alpha . \mathbf{N}) \in \Phi$, then $\Phi * \neg([w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N}) \in \Gamma_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

$\nabla_{\mathfrak{f}}$ If $\neg(\mathbf{G} \doteq^{\alpha \to \beta} \mathbf{H}) \in \Phi$, then $\Phi * \neg(\mathbf{G}w \doteq^{\beta} \mathbf{H}w) \in \Gamma_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

©Benzmüller & Brown, 2006 UNIVERSITÄT DES SAARLANDES SEMHOL'06-[4] – p.191

Extens. Abstract Consistency Properties

Let $\mathfrak{I}_\Sigma$ be a class of sets of Σ-sentences. We define (where $\Phi \in \mathfrak{I}_\Sigma$, $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{G}, \mathbf{H}, (\lambda X_\alpha \mathbf{.} \mathbf{M}), (\lambda X_\alpha \mathbf{.} \mathbf{N}) \in cwff_{\alpha\to\beta}(\Sigma)$ are arbitrary):

$\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \mathfrak{I}_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \mathfrak{I}_\Sigma$.

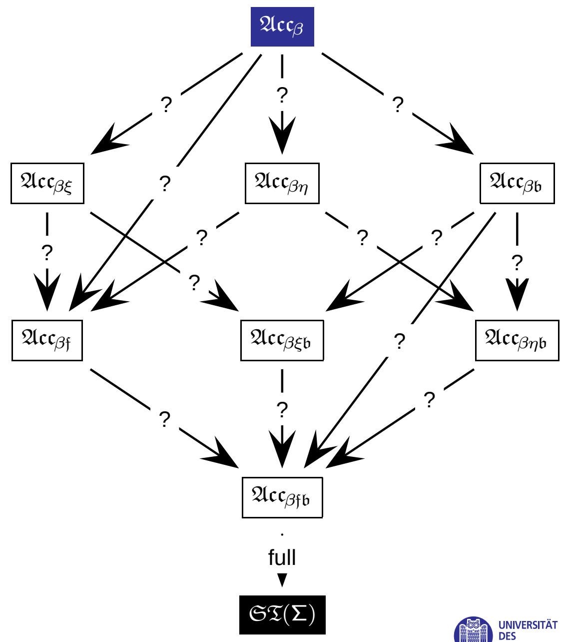
∇_η If $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \mathfrak{I}_\Sigma$.

∇_ξ If $\neg(\lambda X_\alpha \mathbf{.} \mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha \mathbf{.} \mathbf{N}) \in \Phi$, then $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \mathfrak{I}_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

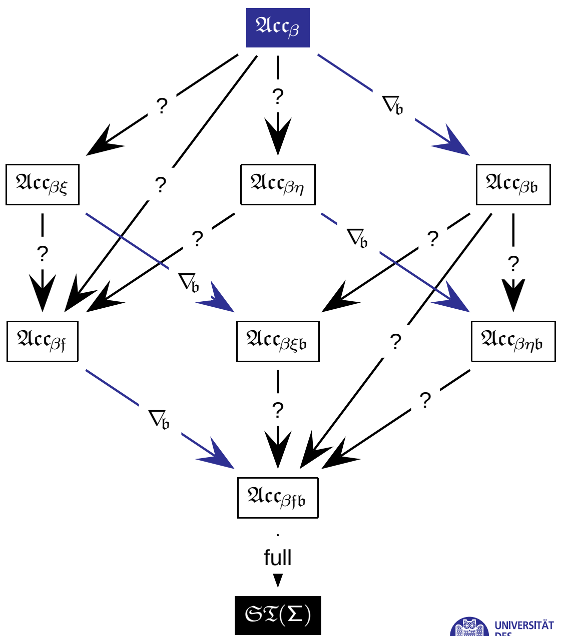
$\nabla_{\mathfrak{f}}$ If $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \Phi$, then $\Phi * \neg(\mathbf{G}w \doteq^\beta \mathbf{H}w) \in \mathfrak{I}_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

(These properties are new in [Chris-PhD-99,Chad-PhD-04,JSL-04])

Extens. Abstract Consistency Properties



Extens. Abstract Consistency Properties



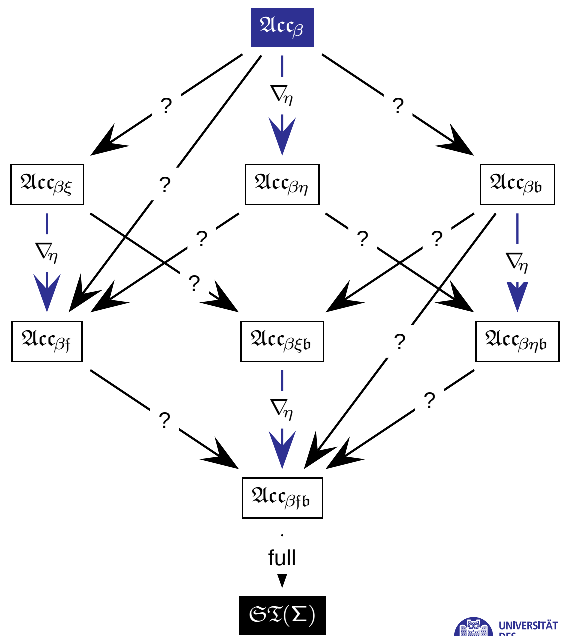
Basic AC Properties for $\mathfrak{Acc}_\beta$

∇_c	...	$\nabla_\vee$	...
$\nabla_\neg$	...	$\nabla_\wedge$	...
∇_β	...	$\nabla_\forall$	...
		$\nabla_\exists$	...

Extens. AC Properties

$\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \mathfrak{I}_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \mathfrak{I}_\Sigma$.

Extens. Abstract Consistency Properties



Basic AC Properties for $\mathfrak{Acc}_\beta$

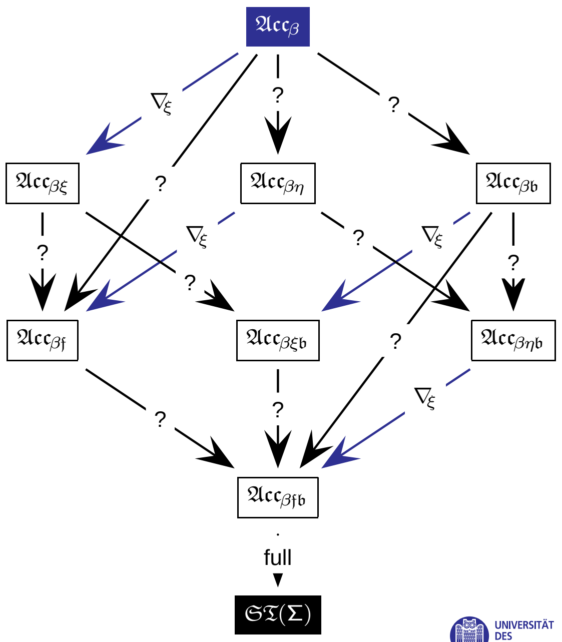
∇_c	...	$\nabla_\vee$	...
$\nabla_\neg$	...	$\nabla_\wedge$	...
∇_β	...	$\nabla_\forall$	...
		$\nabla_\exists$	...

Extens. AC Properties

$\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \mathfrak{I}_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \mathfrak{I}_\Sigma$.

∇_η If $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \mathfrak{I}_\Sigma$.

Extens. Abstract Consistency Properties



Basic AC Properties for $\mathfrak{Acc}_\beta$

∇_c	...	$\nabla_\vee$	...
$\nabla_\neg$	...	$\nabla_\wedge$	...
∇_β	...	$\nabla_\forall$	...
		$\nabla_\exists$	...

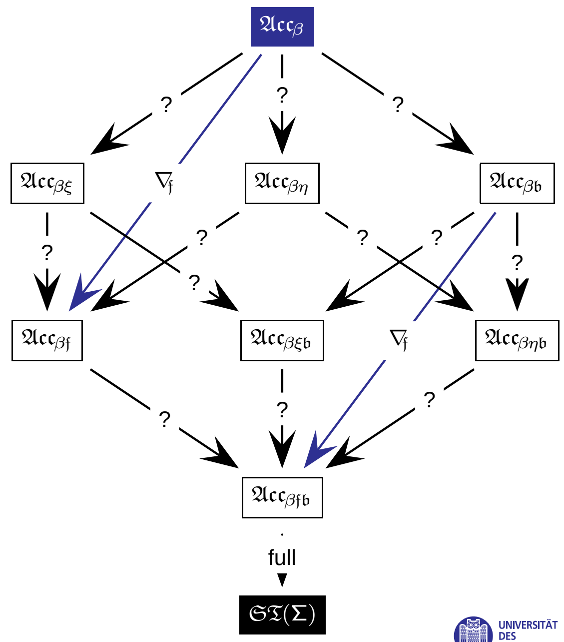
Extens. AC Properties

$\nabla_\mathfrak{b}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$.

∇_η If $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

∇_ξ If $\neg(\lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha.\mathbf{N}) \in \Phi$, then $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \Gamma_\Sigma$ for any new $w_\alpha \in \Sigma_\alpha$.

Extens. Abstract Consistency Properties



Basic AC Properties for $\mathfrak{Acc}_\beta$

∇_c	...	$\nabla_\vee$	...
$\nabla_\neg$	...	$\nabla_\wedge$	...
∇_β	...	$\nabla_\forall$	...
		$\nabla_\exists$	...

Extens. AC Properties

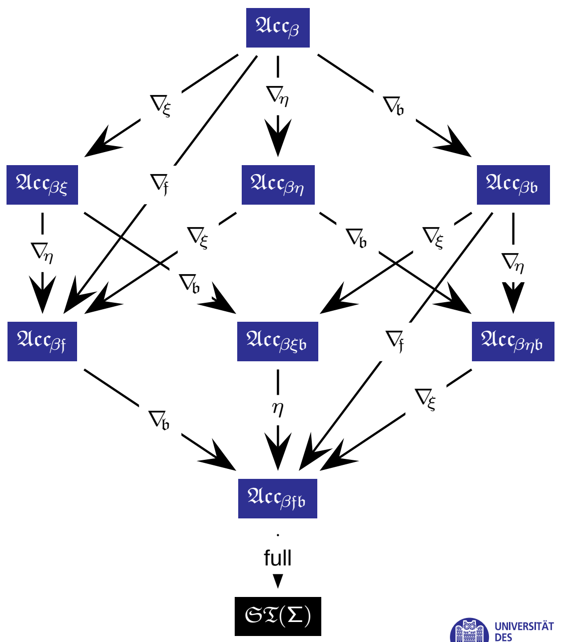
$\nabla_\mathfrak{b}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$.

∇_η If $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

∇_ξ If $\neg(\lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha.\mathbf{N}) \in \Phi$, then $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \Gamma_\Sigma$ for any new $w_\alpha \in \Sigma_\alpha$.

$\nabla_\mathfrak{f}$ If $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \Phi$, then $\Phi * \neg(\mathbf{G}w \doteq^\beta \mathbf{H}w) \in \Gamma_\Sigma$ for any new $w_\alpha \in \Sigma_\alpha$.

Extens. Abstract Consistency Properties



Basic AC Properties for $\mathfrak{Acc}_\beta$

∇_c	...	$\nabla_\vee$	...
$\nabla_\neg$	...	$\nabla_\wedge$	...
∇_β	...	$\nabla_\forall$	...
		$\nabla_\exists$	...

Extens. AC Properties

$\nabla_\mathfrak{b}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$.

∇_η If $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

∇_ξ If $\neg(\lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha.\mathbf{N}) \in \Phi$, then $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \Gamma_\Sigma$ for any new $w_\alpha \in \Sigma_\alpha$.

$\nabla_\mathfrak{f}$ If $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \Phi$, then $\Phi * \neg(\mathbf{G}w \doteq^\beta \mathbf{H}w) \in \Gamma_\Sigma$ for any new $w_\alpha \in \Sigma_\alpha$.

Abstract Consistency Class $\mathfrak{Acc}_\beta$

Defn.: (Contd.) Let Σ be a signature and Γ_Σ be a class of sets of Σ-sentences that is closed under subsets.

Abstract Consistency Class $\mathfrak{Acc}_\beta$

Defn.: (Contd.) Let Σ be a signature and Γ_Σ be a class of sets of Σ-sentences that is closed under subsets.

- If $\nabla_c, \nabla_\neg, \nabla_\beta, \nabla_\vee, \nabla_\wedge, \nabla_\forall$ and $\nabla_\exists$ are valid for Γ_Σ, then Γ_Σ is called an abstract consistency class for Σ-models.

Abstract Consistency Class $\mathfrak{Acc}_\beta$

Defn.: (Contd.) Let Σ be a signature and Γ_Σ be a class of sets of Σ-sentences that is closed under subsets.

- If $\nabla_c, \nabla_\neg, \nabla_\beta, \nabla_\vee, \nabla_\wedge, \nabla_\forall$ and $\nabla_\exists$ are valid for Γ_Σ, then Γ_Σ is called an abstract consistency class for Σ-models.
- We will denote the collection of abstract consistency classes by $\mathfrak{Acc}_\beta$.

Abstract Consistency Class $\mathfrak{Acc}_\beta$

Defn.: (Contd.) Let Σ be a signature and Γ_Σ be a class of sets of Σ-sentences that is closed under subsets.

- If $\nabla_c, \nabla_\neg, \nabla_\beta, \nabla_\vee, \nabla_\wedge, \nabla_\forall$ and $\nabla_\exists$ are valid for Γ_Σ, then Γ_Σ is called an abstract consistency class for Σ-models.
- We will denote the collection of abstract consistency classes by $\mathfrak{Acc}_\beta$.
- Similarly, we introduce the following collections of specialized abstract consistency classes (with primitive equality): $\mathfrak{Acc}_{\beta\eta}, \mathfrak{Acc}_{\beta\xi}, \mathfrak{Acc}_{\beta\mathfrak{f}}, \mathfrak{Acc}_{\beta\mathfrak{b}}, \mathfrak{Acc}_{\beta\eta\mathfrak{b}}, \mathfrak{Acc}_{\beta\xi\mathfrak{b}}, \mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$, where we indicate by indices which additional properties from $\{\nabla_\eta, \nabla_\xi, \nabla_\mathfrak{f}, \nabla_\mathfrak{b}\}$ are required.

Ex.: Abstract Consistency Class

- not an abstract consistency class: $\{\{\neg(A \vee B), \neg A\}, \{\neg(A \vee B)\}, \{\neg A\}, \{\}\}$

Ex.: Abstract Consistency Class

- not an abstract consistency class:
  $\{\{\neg(A \vee B), \neg A\}, \{\neg(A \vee B)\}, \{\neg A\}, \{\}\}$
- still not:
  $\{\{\neg(A \vee B), \neg A, \neg B\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B)\}, \{\neg A\}, \{\}\}$

Ex.: Abstract Consistency Class

- not an abstract consistency class:
  $\{\{\neg(A \vee B), \neg A\}, \{\neg(A \vee B)\}, \{\neg A\}, \{\}\}$
- still not:
  $\{\{\neg(A \vee B), \neg A, \neg B\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B)\}, \{\neg A\}, \{\}\}$
- how about this one:
  $\Gamma := \{\{\neg(A \vee B), \neg A, \neg B\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B), \neg B\}, \{\neg A, \neg B\}, \{\neg(A \vee B)\}, \{\neg A\}, \{\neg B\}, \{\}\}$

Ex.: Abstract Consistency Class

- not an abstract consistency class:
  $\{\{\neg(A \vee B), \neg A\}, \{\neg(A \vee B)\}, \{\neg A\}, \{\}\}$
- still not:
  $\{\{\neg(A \vee B), \neg A, \neg B\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B)\}, \{\neg A\}, \{\}\}$
- how about this one:
  $\Gamma := \{\{\neg(A \vee B), \neg A, \neg B\}, \{\neg(A \vee B), \neg A\}, \{\neg(A \vee B), \neg B\}, \{\neg A, \neg B\}, \{\neg(A \vee B)\}, \{\neg A\}, \{\neg B\}, \{\}\}$
- and how about this:
  $\Gamma_0 := \Gamma$
  $\Phi \in \Gamma_i \wedge A \in \Phi \wedge B =_{\beta\eta} A \wedge B \neq A \wedge (\Phi * B) \notin \Gamma_i \longrightarrow$
  $\Gamma_{i+1} := \text{close-under-subsets}(\Gamma_i * (\Phi * B))$
  $\Gamma^* := \Gamma_\infty$

Rem.: Possible Generalization

The work presented here is based on the choice of the primitive logical connectives $\neg, \vee$ and Π^α.

Rem.: Possible Generalization

The work presented here is based on the choice of the primitive logical connectives $\neg, \vee$ and Π^α. A means to generalize the framework over the concrete choice of logical primitives is provided by the uniform notation approach as, for instance, given in [Fitting96].

Rem.: Possible Generalization

The work presented here is based on the choice of the primitive logical connectives $\neg, \vee$ and Π^α. A means to generalize the framework over the concrete choice of logical primitives is provided by the uniform notation approach as, for instance, given in [Fitting96]. This can be done in straightforward manner: $\nabla_\wedge$ becomes an α-property, $\nabla_\vee$ becomes a β-property, $\nabla_\forall$ becomes a γ-property, and $\nabla_\exists$ becomes a δ-property.

Rem.: Possible Generalization

The work presented here is based on the choice of the primitive logical connectives $\neg, \vee$ and Π^α. A means to generalize the framework over the concrete choice of logical primitives is provided by the uniform notation approach as, for instance, given in [Fitting96]. This can be done in straightforward manner: $\nabla_\wedge$ becomes an α-property, $\nabla_\vee$ becomes a β-property, $\nabla_\forall$ becomes a γ-property, and $\nabla_\exists$ becomes a δ-property. Thus they will have the following form:

Rem.: Possible Generalization

The work presented here is based on the choice of the primitive logical connectives $\neg, \vee$ and Π^α. A means to generalize the framework over the concrete choice of logical primitives is provided by the uniform notation approach as, for instance, given in [Fitting96]. This can be done in straightforward manner: $\nabla_\wedge$ becomes an α-property, $\nabla_\vee$ becomes a β-property, $\nabla_\forall$ becomes a γ-property, and $\nabla_\exists$ becomes a δ-property. Thus they will have the following form:

α-case If $\alpha \in \Phi$, then $\Phi * \alpha_1 * \alpha_2 \in \Gamma_\Sigma$.

Rem.: Possible Generalization

The work presented here is based on the choice of the primitive logical connectives $\neg, \vee$ and Π^α. A means to generalize the framework over the concrete choice of logical primitives is provided by the uniform notation approach as, for instance, given in [Fitting96]. This can be done in straightforward manner: $\nabla_\wedge$ becomes an α-property, $\nabla_\vee$ becomes a β-property, $\nabla_\forall$ becomes a γ-property, and $\nabla_\exists$ becomes a δ-property. Thus they will have the following form:

α-case If $\alpha \in \Phi$, then $\Phi * \alpha_1 * \alpha_2 \in \Gamma_\Sigma$.

β-case If $\beta \in \Phi$, then $\Phi * \beta_1 \in \Gamma_\Sigma$ or $\Phi * \beta_2 \in \Gamma_\Sigma$.

Rem.: Possible Generalization

The work presented here is based on the choice of the primitive logical connectives $\neg, \vee$ and Π^α. A means to generalize the framework over the concrete choice of logical primitives is provided by the uniform notation approach as, for instance, given in [Fitting96]. This can be done in straightforward manner: $\nabla_\wedge$ becomes an α-property, $\nabla_\vee$ becomes a β-property, $\nabla_\forall$ becomes a γ-property, and $\nabla_\exists$ becomes a δ-property. Thus they will have the following form:

α-case If $\alpha \in \Phi$, then $\Phi * \alpha_1 * \alpha_2 \in \Gamma_\Sigma$.

β-case If $\beta \in \Phi$, then $\Phi * \beta_1 \in \Gamma_\Sigma$ or $\Phi * \beta_2 \in \Gamma_\Sigma$.

γ-case If $\gamma \in \Phi$, then $\Phi * \gamma\mathbf{W} \in \Gamma_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

Rem.: Possible Generalization

The work presented here is based on the choice of the primitive logical connectives $\neg, \vee$ and Π^α. A means to generalize the framework over the concrete choice of logical primitives is provided by the uniform notation approach as, for instance, given in [Fitting96]. This can be done in straightforward manner: $\nabla_\wedge$ becomes an α-property, $\nabla_\vee$ becomes a β-property, $\nabla_\forall$ becomes a γ-property, and $\nabla_\exists$ becomes a δ-property. Thus they will have the following form:

α-case If $\alpha \in \Phi$, then $\Phi * \alpha_1 * \alpha_2 \in \Gamma_\Sigma$.

β-case If $\beta \in \Phi$, then $\Phi * \beta_1 \in \Gamma_\Sigma$ or $\Phi * \beta_2 \in \Gamma_\Sigma$.

γ-case If $\gamma \in \Phi$, then $\Phi * \gamma\mathbf{W} \in \Gamma_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

δ-case If $\delta \in \Phi$, then $\Phi * \delta\mathrm{w} \in \Gamma_\Sigma$ for any parameter $\mathrm{w}_\alpha \in \Sigma$ which does not occur in any sentence of Φ.

Def.: Sufficiently Σ-Pure

We introduce a technical side-condition that ensures that we always have enough witness constants.

Def.: Sufficiently Σ-Pure

We introduce a technical side-condition that ensures that we always have enough witness constants.

> Let Σ be a signature and Φ be a set of Σ-sentences. Φ is called sufficiently Σ-pure if for each type α there is a set $\mathcal{P}_\alpha \subseteq \Sigma_\alpha$ of parameters with equal cardinality to $wff_\alpha(\Sigma)$, such that the elements of $\mathcal{P}_\alpha$ do not occur in the sentences of Φ.

Def.: Sufficiently Σ-Pure

We introduce a technical side-condition that ensures that we always have enough witness constants.

> Let Σ be a signature and Φ be a set of Σ-sentences. Φ is called sufficiently Σ-pure if for each type α there is a set $\mathcal{P}_\alpha \subseteq \Sigma_\alpha$ of parameters with equal cardinality to $wff_\alpha(\Sigma)$, such that the elements of $\mathcal{P}_\alpha$ do not occur in the sentences of Φ.

This can be obtained in practice by enriching the signature with spurious parameters.

Saturation

Defn.: Consider the following property (where $\Phi \in \Gamma_\Sigma$, $\mathbf{A} \in cwff_o(\Sigma)$):

Saturation

Defn.: Consider the following property (where $\Phi \in \Gamma_\Sigma$, $\mathbf{A} \in cwff_o(\Sigma)$):

∇_{sat} Either $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} \in \Gamma_\Sigma$.

Saturation

Defn.: Consider the following property (where $\Phi \in \Gamma_\Sigma$, $\mathbf{A} \in cwff_o(\Sigma)$):

∇_{sat} Either $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} \in \Gamma_\Sigma$.

- We call an abstract consistency class Γ_Σ saturated if ∇_{sat} holds for all A.

Ex.: Saturated

- consider Γ (and Γ^*) from before: $\{\{\neg(A \lor B), \neg A, \neg B\}, \{\neg(A \lor B), \neg A\}, \{\neg(A \lor B), \neg B\}, \{\neg A, \neg B\}, \{\neg(A \lor B)\}, \{\neg A\}, \{\neg B\}, \{\}\}$

Ex.: Saturated

- consider Γ (and Γ^*) from before: $\{\{\neg(A \lor B), \neg A, \neg B\}, \{\neg(A \lor B), \neg A\}, \{\neg(A \lor B), \neg B\}, \{\neg A, \neg B\}, \{\neg(A \lor B)\}, \{\neg A\}, \{\neg B\}, \{\}\}$
- Γ (and Γ^*) is not saturated: for instance, it does not provide information on the formulas $(\neg A \lor B) \lor A$ and $\Pi^o(\lambda X_o.X)$

Def.: Saturated Extension

Def.: (Saturated Extension)

Let $\Gamma_\Sigma, \Gamma'_\Sigma \in \mathfrak{Acc}_*$ be abstract consistency classes. We say Γ'_Σ is an extension of Γ_Σ if $\Phi \in \Gamma'_\Sigma$ for every (sufficiently Σ-pure) $\Phi \in \Gamma_\Sigma$. We say Γ'_Σ is a saturated extension of Γ_Σ if Γ'_Σ is saturated and an extension of Γ_Σ.

Ex.: ACC without Saturated Extension

There exist abstract consistency classes Γ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ which have no saturated extension.

Ex.: ACC without Saturated Extension

There exist abstract consistency classes Γ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ which have no saturated extension.

Example:

Ex.: ACC without Saturated Extension

There exist abstract consistency classes Γ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ which have no saturated extension.

Example:

Let $a_o, b_o, q_{o\to o} \in \Sigma$ and $\Phi := \{a, b, (qa), \neg(qb)\}$. We construct an abstract consistency class Γ_Σ from Φ by first building the closure Φ' of Φ under relation $=_\beta$ and then taking the power set of Φ'. It is easy to check that this Γ_Σ is in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$.

Ex.: ACC without Saturated Extension

There exist abstract consistency classes Γ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ which have no saturated extension.

Example:
Let $a_o, b_o, q_{o\to o} \in \Sigma$ and $\Phi := \{a, b, (qa), \neg(qb)\}$. We construct an abstract consistency class Γ_Σ from Φ by first building the closure Φ' of Φ under relation $=_\beta$ and then taking the power set of Φ'. It is easy to check that this Γ_Σ is in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$. Suppose we have a saturated extension Γ'_Σ of Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$.

Ex.: ACC without Saturated Extension

There exist abstract consistency classes Γ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ which have no saturated extension.

Example:
Let $a_o, b_o, q_{o\to o} \in \Sigma$ and $\Phi := \{a, b, (qa), \neg(qb)\}$. We construct an abstract consistency class Γ_Σ from Φ by first building the closure Φ' of Φ under relation $=_\beta$ and then taking the power set of Φ'. It is easy to check that this Γ_Σ is in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$. Suppose we have a saturated extension Γ'_Σ of Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$. Then $\Phi \in \Gamma'_\Sigma$ since Φ is finite (hence sufficiently pure) .

Ex.: ACC without Saturated Extension

There exist abstract consistency classes Γ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ which have no saturated extension.

Example:
Let $a_o, b_o, q_{o\to o} \in \Sigma$ and $\Phi := \{a, b, (qa), \neg(qb)\}$. We construct an abstract consistency class Γ_Σ from Φ by first building the closure Φ' of Φ under relation $=_\beta$ and then taking the power set of Φ'. It is easy to check that this Γ_Σ is in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$. Suppose we have a saturated extension Γ'_Σ of Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$. Then $\Phi \in \Gamma'_\Sigma$ since Φ is finite (hence sufficiently pure) . By saturation, $\Phi * (a \doteq^o b) \in \Gamma'_\Sigma$ or $\Phi * \neg(a \doteq^o b) \in \Gamma'_\Sigma$.

Ex.: ACC without Saturated Extension

There exist abstract consistency classes Γ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ which have no saturated extension.

Example:
Let $a_o, b_o, q_{o\to o} \in \Sigma$ and $\Phi := \{a, b, (qa), \neg(qb)\}$. We construct an abstract consistency class Γ_Σ from Φ by first building the closure Φ' of Φ under relation $=_\beta$ and then taking the power set of Φ'. It is easy to check that this Γ_Σ is in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$. Suppose we have a saturated extension Γ'_Σ of Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$. Then $\Phi \in \Gamma'_\Sigma$ since Φ is finite (hence sufficiently pure) . By saturation, $\Phi * (a \doteq^o b) \in \Gamma'_\Sigma$ or $\Phi * \neg(a \doteq^o b) \in \Gamma'_\Sigma$. In the first case, applying $\nabla_\forall$ with the constant q, ∇_β, $\nabla_\vee$ and ∇_c contradicts $(qa), \neg(qb) \in \Phi$.

Ex.: ACC without Saturated Extension

There exist abstract consistency classes Γ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ which have no saturated extension.

Example:
Let $a_o, b_o, q_{o\to o} \in \Sigma$ and $\Phi := \{a, b, (qa), \neg(qb)\}$. We construct an abstract consistency class Γ_Σ from Φ by first building the closure Φ' of Φ under relation $=_\beta$ and then taking the power set of Φ'. It is easy to check that this Γ_Σ is in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$. Suppose we have a saturated extension Γ'_Σ of Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$. Then $\Phi \in \Gamma'_\Sigma$ since Φ is finite (hence sufficiently pure) . By saturation, $\Phi * (a \doteq^o b) \in \Gamma'_\Sigma$ or $\Phi * \neg(a \doteq^o b) \in \Gamma'_\Sigma$. In the first case, applying $\nabla_\forall$ with the constant q, ∇_β, $\nabla_\vee$ and ∇_c contradicts $(qa), \neg(qb) \in \Phi$. In the second case, $\nabla_\mathfrak{b}$ and ∇_c contradict $a, b \in \Phi$.

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated abstract consistency class and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.
For all $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ we have:

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated abstract consistency class and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.
For all $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated abstract consistency class and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.
For all $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.
- Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$ and $wff_\alpha(\Sigma)$)

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated abstract consistency class and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.
For all $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.
- Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$ and $wff_\alpha(\Sigma)$)

Proof: (Sketch)

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated abstract consistency class and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.
For all $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.
- Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$ and $wff_\alpha(\Sigma)$)

Proof: (Sketch)

...not yet ...

Model Existence for Henkin Models

Thm.: Let Γ_Σ be a saturated abstract consistency class in $\mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$ and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.

Model Existence for Henkin Models

Thm.: Let Γ_Σ be a saturated abstract consistency class in $\mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$ and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.

- Then there is a Henkin Model that satisfies Φ.

Model Existence for Henkin Models

Thm.: Let Γ_Σ be a saturated abstract consistency class in $\mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$ and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.

- Then there is a Henkin Model that satisfies Φ.
- Furthermore, each domain of the model has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$ and $wff_\alpha(\Sigma)$).

Model Existence for Henkin Models

Thm.: Let Γ_Σ be a saturated abstract consistency class in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.

- Then there is a Henkin Model that satisfies Φ.
- Furthermore, each domain of the model has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $\mathit{wff}_\alpha(\Sigma)$ and $\mathit{wff}_\alpha(\Sigma)$).

Proof: (Sketch)

Model Existence for Henkin Models

Thm.: Let Γ_Σ be a saturated abstract consistency class in $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences.

- Then there is a Henkin Model that satisfies Φ.
- Furthermore, each domain of the model has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $\mathit{wff}_\alpha(\Sigma)$ and $\mathit{wff}_\alpha(\Sigma)$).

Proof: (Sketch)

...not yet ...

Completeness of $\mathfrak{NK}_*$ via Abstract Consistency

$\mathfrak{NK}_*$-Consistent/Inconsistent

Def.: A set of sentences Φ is $\mathfrak{NK}_*$-inconsistent if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{F}_o$, and $\mathfrak{NK}_*$-consistent otherwise.

$\mathfrak{NK}_*$-Consistent/Inconsistent

Def.: A set of sentences Φ is $\mathfrak{NK}_*$-inconsistent if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{F}_o$, and $\mathfrak{NK}_*$-consistent otherwise.

- We will now consider the class

$$\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \text{ is } \mathfrak{NK}_*\text{-consistent}\}$$

$\mathfrak{NK}_*$-Consistent/Inconsistent

Def.: A set of sentences Φ is $\mathfrak{NK}_*$-inconsistent if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{F}_o$, and $\mathfrak{NK}_*$-consistent otherwise.

- We will now consider the class

$$\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \text{ is } \mathfrak{NK}_*\text{-consistent}\}$$

- i.e.

$$\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We first show: Γ_Σ^* is closed under subsets)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We first show: Γ^*_Σ is closed under subsets)

Obviously Γ^*_Σ is closed under subsets, since any subset of an $\mathfrak{NK}_*$-consistent set is $\mathfrak{NK}_*$-consistent. (If $\Psi \subseteq \Phi$ and $\Psi \Vdash_{\mathfrak{NK}_*} \mathbf{F}_o$ then clearly $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{F}_o$)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We now show: $\nabla_c, \nabla_\neg, \nabla_\beta, \nabla_\vee, \nabla_\wedge, \nabla_\forall, \nabla_\eta, \nabla_\xi, \nabla_\mathfrak{f}, \nabla_\mathfrak{b}, \nabla_{sat}$)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We now show: $\nabla_c, \nabla_\neg, \nabla_\beta, \nabla_\vee, \nabla_\wedge, \nabla_\forall, \nabla_\eta, \nabla_\xi, \nabla_\mathfrak{f}, \nabla_\mathfrak{b}, \nabla_{sat}$)

We now check the remaining conditions. We prove all the properties by proving their contrapositive.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_c)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_c)

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_c)

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

Suppose $\mathbf{A}, \neg\mathbf{A} \in \Phi$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_c)

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

Suppose $\mathbf{A}, \neg\mathbf{A} \in \Phi$.

$$\dfrac{\dfrac{}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Hyp) \qquad \dfrac{}{\Phi \Vdash \neg\mathbf{A}}\,\mathfrak{NK}(Hyp)}{\Phi \Vdash \mathbf{F}_o}\,\mathfrak{NK}(\neg E)$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_c)

∇_c If $\mathbf{A}$ is atomic, then $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.

Suppose $\mathbf{A}, \neg\mathbf{A} \in \Phi$.

$$\dfrac{\dfrac{}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(Hyp) \qquad \dfrac{}{\Phi \Vdash \neg\mathbf{A}}\,\mathfrak{NK}(Hyp)}{\Phi \Vdash \mathbf{F}_o}\,\mathfrak{NK}(\neg E)$$

Hence Φ is $\mathfrak{NK}_*$-inconsistent.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq \mathit{cwff}_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_β)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq \mathit{cwff}_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_β)

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma^*_\Sigma$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq \mathit{cwff}_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_β)

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma^*_\Sigma$.

Let $\mathbf{A} \in \Phi$, $\mathbf{A} =_\beta \mathbf{B}$ and $\Phi * \mathbf{B}$ be $\mathfrak{NK}_*$-inconsistent.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq \mathit{cwff}_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_β)

∇_β If $\mathbf{A} =_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma^*_\Sigma$.

Let $\mathbf{A} \in \Phi$, $\mathbf{A} =_\beta \mathbf{B}$ and $\Phi * \mathbf{B}$ be $\mathfrak{NK}_*$-inconsistent. That is, $\Phi * \mathbf{B} \Vdash \mathbf{F}_o$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_β)

∇_β If $\mathbf{A} {=_\beta} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma^*_\Sigma$.

Let $\mathbf{A} \in \Phi$, $\mathbf{A} {=_\beta} \mathbf{B}$ and $\Phi * \mathbf{B}$ be $\mathfrak{NK}_*$-inconsistent. That is, $\Phi * \mathbf{B} \Vdash \mathbf{F}_o$.

$$\boxed{\nabla_\beta} \quad \dfrac{\dfrac{\Phi * B \Vdash F_o}{\Phi \Vdash \neg B}\,nk(\neg I) \qquad \dfrac{\dfrac{}{\Phi \Vdash A}\,nk(Hyp)}{\Phi \Vdash B}\,nk(\beta)}{\Phi \Vdash F_o}\,nk(\neg E)$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\neg$)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\neg$)

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma^*_\Sigma$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\neg$)

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma^*_\Sigma$.

Suppose $\neg\neg\mathbf{A} \in \Phi$ and $\Phi * \mathbf{A}$ is $\mathfrak{NK}_*$-inconsistent.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\neg$)

$\nabla_\neg$ If $\neg\neg\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma^*_\Sigma$.

Suppose $\neg\neg\mathbf{A} \in \Phi$ and $\Phi * \mathbf{A}$ is $\mathfrak{NK}_*$-inconsistent.

$$\boxed{\nabla_\neg}\qquad \dfrac{\dfrac{\Phi * A \Vdash F_o}{\Phi \Vdash \neg A}\,nk(\neg I) \qquad \dfrac{}{\Phi \Vdash \neg\neg A}\,nk(Hyp)}{\Phi \Vdash F_o}\,nk(\neg E)$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\vee$)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\vee$)

$\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma^*_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma^*_\Sigma$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\vee$)

$\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma^*_\Sigma$ or $\Phi * \mathbf{B} \in \Gamma^*_\Sigma$.

Suppose $(\mathbf{A} \vee \mathbf{B}) \in \Phi$ and both $\Phi * \mathbf{A}$ and $\Phi * \mathbf{B}$ are $\mathfrak{NK}_*$-inconsistent.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\vee$)

$\nabla_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \in \Gamma_\Sigma^*$ or $\Phi * \mathbf{B} \in \Gamma_\Sigma^*$.

Suppose $(\mathbf{A} \vee \mathbf{B}) \in \Phi$ and both $\Phi * \mathbf{A}$ and $\Phi * \mathbf{B}$ are $\mathfrak{NK}_*$-inconsistent.

$\boxed{\nabla_\vee}$

$$\dfrac{\dfrac{}{\Phi \Vdash A \vee B}\,nk(Hyp) \qquad \Phi * A \Vdash F_o \qquad \Phi * B \Vdash F_o}{\Phi \Vdash F_o}\,nk(\vee E)$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi\Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\wedge$)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi\Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\wedge$)

$\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma^*$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi\Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\wedge$)

$\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma^*$.

Suppose $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$ and $\Phi * \neg\mathbf{A} * \neg\mathbf{B}$ is $\mathfrak{NK}_*$-inconsistent.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi\Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\wedge$)

$\nabla_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, then $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma^*_\Sigma$.

Suppose $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$ and $\Phi * \neg\mathbf{A} * \neg\mathbf{B}$ is $\mathfrak{NK}_*$-inconsistent.

$$\dfrac{\dfrac{\dfrac{\Phi * \neg A * \neg B \Vdash F_o}{\Phi * \neg A \Vdash B}\,nk(Contr)}{\Phi * \neg A \Vdash A \vee B}\,nk(\vee I_R) \qquad \dfrac{}{\Phi * \neg A \Vdash \neg(A \vee B)}\,nk(Hyp)}{\Phi * \neg A \Vdash F_o}\,nk(\neg E)$$

$$\dfrac{\dfrac{\dfrac{\Phi * \neg A \Vdash F_o}{\Phi \Vdash A}\,nk(Contr)}{\Phi \Vdash A \vee B}\,nk(\vee I_L) \qquad \dfrac{}{\Phi \Vdash \neg(A \vee B)}\,nk(Hyp)}{\Phi \Vdash F_o}\,nk(\neg E)$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\forall$)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\forall$)

$\nabla_\forall$ If $\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \mathbf{F}\mathbf{W} \in \Gamma^*_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\forall$)

$\nabla_\forall$ If $\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \mathbf{F}\mathbf{W} \in \Gamma^*_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

Suppose $(\Pi^\alpha\mathbf{G}) \in \Phi$ and $\Phi * (\mathbf{G}\mathbf{A})$ is $\mathfrak{NK}_*$-inconsistent.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\forall$)

$\nabla_\forall$ If $\Pi^\alpha \mathbf{F} \in \Phi$, then $\Phi * \mathbf{FW} \in \Gamma^*_\Sigma$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

Suppose $(\Pi^\alpha \mathbf{G}) \in \Phi$ and $\Phi * (\mathbf{GA})$ is $\mathfrak{NK}_*$-inconsistent.

$$\boxed{\nabla_\forall} \qquad \dfrac{\dfrac{}{\Phi \Vdash \neg(GA)}\,nk(Hyp) \qquad \dfrac{\dfrac{}{\Phi \Vdash \Pi^\alpha G}\,nk(Hyp)}{\Phi \Vdash (GA)}\,nk(\Pi E)}{\Phi \Vdash F_o}\,nk(\neg E)$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\exists$)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\exists$)

$\nabla_\exists$ If $\neg\Pi^\alpha \mathbf{F} \in \Phi$, then $\Phi * \neg(\mathbf{F}\mathrm{w}) \in \Gamma^*_\Sigma$ for any parameter $\mathrm{w}_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\exists$)

$\nabla_\exists$ If $\neg\Pi^\alpha \mathbf{F} \in \Phi$, then $\Phi * \neg(\mathbf{F}\mathrm{w}) \in \Gamma^*_\Sigma$ for any parameter $\mathrm{w}_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Suppose $\neg(\Pi^\alpha \mathbf{G}) \in \Phi$, w_α is a parameter which does not occur in Φ, and $\Phi * \neg(\mathbf{G}\mathrm{w})$ is $\mathfrak{NK}_*$-inconsistent.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\exists$)

$\nabla_\exists$ If $\neg\Pi^\alpha\mathbf{F} \in \Phi$, then $\Phi * \neg(\mathbf{F}w) \in \Gamma^*_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Suppose $\neg(\Pi^\alpha\mathbf{G}) \in \Phi$, w_α is a parameter which does not occur in Φ, and $\Phi * \neg(\mathbf{G}w)$ is $\mathfrak{NK}_*$-inconsistent.

$$\boxed{\nabla_\exists}\quad \dfrac{\dfrac{\dfrac{\Phi * \neg G w_\alpha \Vdash F_o}{\Phi \Vdash G w_\alpha}\,\mathcal{NK}(Contr)}{\Phi \Vdash \Pi^\alpha G}\,\mathcal{NK}(\Pi I)^{w_\alpha} \qquad \dfrac{}{\Phi \Vdash \neg(\Pi^\alpha G)}\,\mathcal{NK}(Hyp)}{\Phi \Vdash F_o}\,\mathcal{NK}(\neg E)$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_{sat})

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_{sat})

∇_{sat} Either $\Phi * \mathbf{A} \in \Gamma^*_\Sigma$ or $\Phi * \neg\mathbf{A} \in \Gamma^*_\Sigma$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_{sat})

∇_{sat} Either $\Phi * \mathbf{A} \in \Gamma^*_\Sigma$ or $\Phi * \neg\mathbf{A} \in \Gamma^*_\Sigma$.

Let $\Phi * \mathbf{A}$ and $\Phi * \neg\mathbf{A}$ be $\mathfrak{NK}_*$-inconsistent.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_{sat})

∇_{sat} Either $\Phi * \mathbf{A} \in \Gamma_\Sigma^*$ or $\Phi * \neg\mathbf{A} \in \Gamma_\Sigma^*$.

Let $\Phi * \mathbf{A}$ and $\Phi * \neg\mathbf{A}$ be $\mathfrak{NK}_*$-inconsistent.

$$\boxed{\nabla_{sat}} \qquad \dfrac{\dfrac{\Phi * A \Vdash F_o}{\Phi \Vdash \neg A}\,nk(\neg I) \qquad \dfrac{\Phi * \neg A \Vdash F_o}{\Phi \Vdash \neg\neg A}\,nk(\neg I)}{\Phi \Vdash F_o}\,nk(\neg E)$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: Thus we have shown that Γ_Σ^β is saturated and in $\mathfrak{Acc}_\beta$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: Thus we have shown that Γ_Σ^β is saturated and in $\mathfrak{Acc}_\beta$.

Now let us check the conditions for the additional properties η, ξ, $\mathfrak{f}$, and $\mathfrak{b}$.

Class of Sets of $\mathfrak{NR}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NR}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: Thus we have shown that Γ_Σ^β is saturated and in $\mathfrak{Acc}_\beta$.

Now let us check the conditions for the additional properties η, ξ, $\mathfrak{f}$, and $\mathfrak{b}$.

Class of Sets of $\mathfrak{NR}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NR}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_η)

Class of Sets of $\mathfrak{NR}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NR}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_η)

∇_η If $\mathbf{A} \overset{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma^*$.

Class of Sets of $\mathfrak{NR}_*$-consistent Formulas

Lemma: $\Gamma_\Sigma^* := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NR}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_η)

∇_η If $\mathbf{A} \overset{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma^*$.

Suppose $*$ includes η, and let $\mathbf{A} \in \Phi$, $\mathbf{A} \overset{\beta\eta}{=} \mathbf{B}$ and $\Phi * \mathbf{B}$ be $\mathfrak{NR}_*$-inconsistent.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_η)

∇_η If $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma^*_\Sigma$.

Suppose $*$ includes η, and let $\mathbf{A} \in \Phi$, $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$ and $\Phi * \mathbf{B}$ be $\mathfrak{NK}_*$-inconsistent.

$$\boxed{\nabla_\eta} \qquad \dfrac{\dfrac{\Phi * B \Vdash F_o}{\Phi \Vdash \neg B}\,nk(\neg I) \qquad \dfrac{\dfrac{}{\Phi \Vdash A}\,nk(Hyp)}{\Phi \Vdash B}\,nk(\eta)}{\Phi \Vdash F_o}\,nk(\neg E)$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_ξ)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_ξ)

∇_ξ If $\neg(\lambda X_\alpha \mathbf{.} \mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha \mathbf{.} \mathbf{N}) \in \Phi$, then $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \Gamma^*_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_ξ)

∇_ξ If $\neg(\lambda X_\alpha \mathbf{.} \mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha \mathbf{.} \mathbf{N}) \in \Phi$, then $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \Gamma^*_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Suppose $*$ includes ξ, $\neg(\lambda X \mathbf{.} \mathbf{M} \doteq^{\alpha\to\beta} \lambda X \mathbf{.} \mathbf{N}) \in \Phi$, and $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N})$ is $\mathfrak{NK}_*$-inconsistent for some parameter w_α which does not occur in any sentence of Φ.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: ∇_ξ)

∇_ξ If $\neg(\lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha.\mathbf{N}) \in \Phi$, then $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \Gamma^*_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Suppose $*$ includes ξ, $\neg(\lambda X.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \Phi$, and $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N})$ is $\mathfrak{NK}_*$-inconsistent for some parameter w_α which does not occur in any sentence of Φ.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Phi * \neg([w/X]M \doteq [w/X]N) \Vdash F_o}{\Phi \Vdash ([w/X]M \doteq [w/X]N)}\,nk(Contr)}{\Phi \Vdash (\lambda X.M \doteq N)w}\,nk(\beta)}{\Phi \Vdash (\forall X.M \doteq N)}\,nk(\Pi I)}{\Phi \Vdash (\lambda X.M \doteq \lambda X.N)}\,nk(\mathfrak{f}) \quad \dfrac{}{\Phi \Vdash \neg(\lambda X.M \doteq \lambda X.N)}\,nk(Hyp)}{\Phi \Vdash F_o}$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\mathfrak{f}$)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\mathfrak{f}$)

$\nabla_\mathfrak{f}$ If $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \Phi$, then $\Phi * \neg(\mathbf{G}w \doteq^\beta \mathbf{H}w) \in \Gamma^*_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\mathfrak{f}$)

$\nabla_\mathfrak{f}$ If $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \Phi$, then $\Phi * \neg(\mathbf{G}w \doteq^\beta \mathbf{H}w) \in \Gamma^*_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Suppose $*$ includes $\mathfrak{f}$, $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \Phi$, and $\Phi * \neg(\mathbf{G}w \doteq^\beta \mathbf{H}w)$ is $\mathfrak{NK}_*$-inconsistent for some parameter w_α which does not occur in any sentence of Φ.

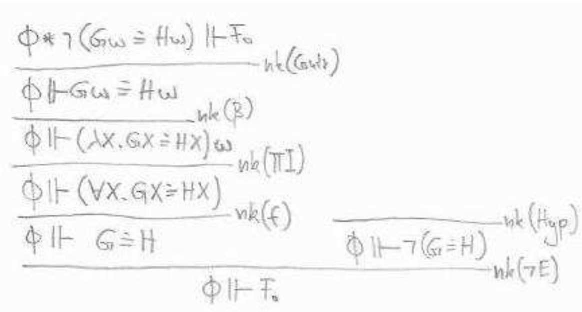
Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\mathfrak{f}$)

$\nabla_\mathfrak{f}$ If $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \Phi$, then $\Phi * \neg(\mathbf{G}w \doteq^{\beta} \mathbf{H}w) \in \Gamma^*_\Sigma$ for any parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in any sentence of Φ.

Suppose $*$ includes $\mathfrak{f}$, $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \Phi$, and $\Phi * \neg(\mathbf{G}w \doteq^{\beta} \mathbf{H}w)$ is $\mathfrak{NK}_*$-inconsistent for some parameter w_α which does not occur in any sentence of Φ.



Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\mathfrak{b}$)

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\mathfrak{b}$)

$\nabla_\mathfrak{b}$ If $\neg(\mathbf{A} \doteq^{o} \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma^*_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma^*_\Sigma$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_\mathfrak{b}$)

$\nabla_\mathfrak{b}$ If $\neg(\mathbf{A} \doteq^{o} \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma^*_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma^*_\Sigma$.

Suppose $*$ includes $\mathfrak{b}$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: (We show: $\nabla_{\mathfrak{b}}$)

$\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma^*_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma^*_\Sigma$.

Suppose $*$ includes $\mathfrak{b}$. Assume that $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$ and that both $\Phi * \neg\mathbf{A} * \mathbf{B}$ and $\Phi * \mathbf{A} * \neg\mathbf{B}$ are $\mathfrak{NK}_*$-inconsistent.

$$\boxed{\nabla_{\mathfrak{b}}} \quad \dfrac{\dfrac{\dfrac{\Phi * A * \neg B \Vdash F_o}{\Phi * A \Vdash B}\,\mathit{hk}(\mathit{Contr}) \quad \dfrac{\Phi * B * \neg A \Vdash F_o}{\Phi * B \Vdash A}\,\mathit{hk}(\mathit{Contr})}{\Phi \Vdash A \doteq B}\,\mathit{hk}(\mathfrak{b}) \quad \dfrac{}{\Phi \Vdash \neg(A \doteq B)}\,\mathit{hk}(\mathit{Hyp})}{\Phi \Vdash F_o}$$

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: Thus, for all $*$ we have Γ^*_Σ is a saturated $\mathfrak{Acc}_*$.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \not\Vdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: Thus, for all $*$ we have Γ^*_Σ is a saturated $\mathfrak{Acc}_*$.

This completes the proof of the lemma. q.e.d.

Class of Sets of $\mathfrak{NK}_*$-consistent Formulas

Lemma: $\Gamma^*_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \Phi \nVdash_{\mathfrak{NK}_*} \mathbf{F}_o\}$ is a saturated $\mathfrak{Acc}_*$.

Proof: Thus, for all $*$ we have Γ^*_Σ is a saturated $\mathfrak{Acc}_*$.

This completes the proof of the lemma. q.e.d.

Henkin's Theorem for $\mathfrak{NK}_*$

Thm.: Let $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$. Every sufficiently Σ-pure $\mathfrak{NK}_*$-consistent set of sentences has an $\mathfrak{M}_*(\Sigma)$-model.

Henkin's Theorem for $\mathfrak{NK}_*$

Thm.: Let $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$. Every sufficiently Σ-pure $\mathfrak{NK}_*$-consistent set of sentences has an $\mathfrak{M}_*(\Sigma)$-model.

Proof: Let Φ be a sufficiently Σ-pure $\mathfrak{NK}_*$-consistent set of sentences.

Henkin's Theorem for $\mathfrak{NK}_*$

Thm.: Let $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$. Every sufficiently Σ-pure $\mathfrak{NK}_*$-consistent set of sentences has an $\mathfrak{M}_*(\Sigma)$-model.

Proof: Let Φ be a sufficiently Σ-pure $\mathfrak{NK}_*$-consistent set of sentences.
By the previous lemma we know that the class of sets of $\mathfrak{NK}_*$-consistent sentences constitute a saturated $\mathfrak{Acc}_*$,

Henkin's Theorem for $\mathfrak{NK}_*$

Thm.: Let $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. Every sufficiently Σ-pure $\mathfrak{NK}_*$-consistent set of sentences has an $\mathfrak{M}_*(\Sigma)$-model.

Proof: Let Φ be a sufficiently Σ-pure $\mathfrak{NK}_*$-consistent set of sentences. By the previous lemma we know that the class of sets of $\mathfrak{NK}_*$-consistent sentences constitute a saturated $\mathfrak{Acc}_*$, thus the Model Existence Theorem guarantees an $\mathfrak{M}_*(\Sigma)$ model for Φ.

Completeness Theorem for $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Completeness Theorem for $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Proof: Let $\mathbf{A}$ be given such that $\mathbf{A}$ is valid in all $\mathfrak{M}_*(\Sigma)$ models that satisfy Φ.

Completeness Theorem for $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Proof: Let $\mathbf{A}$ be given such that $\mathbf{A}$ is valid in all $\mathfrak{M}_*(\Sigma)$ models that satisfy Φ. So, $\Phi * \neg\mathbf{A}$ is unsatisfiable in $\mathfrak{M}_*(\Sigma)$.

Completeness Theorem for $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Proof: Let $\mathbf{A}$ be given such that $\mathbf{A}$ is valid in all $\mathfrak{M}_*(\Sigma)$ models that satisfy Φ. So, $\Phi * \neg\mathbf{A}$ is unsatisfiable in $\mathfrak{M}_*(\Sigma)$. Since only finitely many constants occur in $\neg\mathbf{A}$, $\Phi * \neg\mathbf{A}$ is sufficiently Σ-pure.

Completeness Theorem for $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Proof: Let $\mathbf{A}$ be given such that $\mathbf{A}$ is valid in all $\mathfrak{M}_*(\Sigma)$ models that satisfy Φ. So, $\Phi * \neg\mathbf{A}$ is unsatisfiable in $\mathfrak{M}_*(\Sigma)$. Since only finitely many constants occur in $\neg\mathbf{A}$, $\Phi * \neg\mathbf{A}$ is sufficiently Σ-pure. So, $\Phi * \neg\mathbf{A}$ must be $\mathfrak{NK}_*$-inconsistent by Henkin's theorem above.

Completeness Theorem for $\mathfrak{NK}_*$

Thm.: Let Φ be a sufficiently Σ-pure set of sentences, $\mathbf{A}$ be a sentence, and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathbf{A}$ is valid in all models $\mathcal{M} \in \mathfrak{M}_*(\Sigma)$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Proof: Let $\mathbf{A}$ be given such that $\mathbf{A}$ is valid in all $\mathfrak{M}_*(\Sigma)$ models that satisfy Φ. So, $\Phi * \neg\mathbf{A}$ is unsatisfiable in $\mathfrak{M}_*(\Sigma)$. Since only finitely many constants occur in $\neg\mathbf{A}$, $\Phi * \neg\mathbf{A}$ is sufficiently Σ-pure. So, $\Phi * \neg\mathbf{A}$ must be $\mathfrak{NK}_*$-inconsistent by Henkin's theorem above. Thus, $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$ by $\mathfrak{NK}(Contr)$.

Compactness Theorem for $\mathfrak{NK}_*$

We can use the completeness theorems obtained so far to prove a compactness theorem for $\mathfrak{NK}_*$:

Thm.: Let Φ be a sufficiently Σ-pure set of sentences and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. Φ has an $\mathfrak{M}_*(\Sigma)$-model iff every finite subset of Φ has an $\mathfrak{M}_*(\Sigma)$-model.

Compactness Theorem for $\mathfrak{NK}_*$

We can use the completeness theorems obtained so far to prove a compactness theorem for $\mathfrak{NK}_*$:

Thm.: Let Φ be a sufficiently Σ-pure set of sentences and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. Φ has an $\mathfrak{M}_*(\Sigma)$-model iff every finite subset of Φ has an $\mathfrak{M}_*(\Sigma)$-model.

Proof: (interesting direction by contraposition)

Compactness Theorem for $\mathfrak{NK}_*$

We can use the completeness theorems obtained so far to prove a compactness theorem for $\mathfrak{NK}_*$:

Thm.: Let Φ be a sufficiently Σ-pure set of sentences and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. Φ has an $\mathfrak{M}_*(\Sigma)$-model iff every finite subset of Φ has an $\mathfrak{M}_*(\Sigma)$-model.

Proof: (interesting direction by contraposition)
If Φ has no $\mathfrak{M}_*(\Sigma)$-model, then by the previous Henkin Theorem Φ is $\mathfrak{NK}_*$-inconsistent.

©Benzmüller & Brown, 2006
UNIVERSITÄT DES SAARLANDES
SEMHOL'06-[5] – p.222

Compactness Theorem for $\mathfrak{NK}_*$

We can use the completeness theorems obtained so far to prove a compactness theorem for $\mathfrak{NK}_*$:

Thm.: Let Φ be a sufficiently Σ-pure set of sentences and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. Φ has an $\mathfrak{M}_*(\Sigma)$-model iff every finite subset of Φ has an $\mathfrak{M}_*(\Sigma)$-model.

Proof: (interesting direction by contraposition)
If Φ has no $\mathfrak{M}_*(\Sigma)$-model, then by the previous Henkin Theorem Φ is $\mathfrak{NK}_*$-inconsistent.
Since every $\mathfrak{NK}_*$-proof is finite, this means some finite subset Ψ of Φ is $\mathfrak{NK}_*$-inconsistent.

©Benzmüller & Brown, 2006
UNIVERSITÄT DES SAARLANDES
SEMHOL'06-[5] – p.222

Compactness Theorem for $\mathfrak{NK}_*$

We can use the completeness theorems obtained so far to prove a compactness theorem for $\mathfrak{NK}_*$:

Thm.: Let Φ be a sufficiently Σ-pure set of sentences and $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. Φ has an $\mathfrak{M}_*(\Sigma)$-model iff every finite subset of Φ has an $\mathfrak{M}_*(\Sigma)$-model.

Proof: (interesting direction by contraposition)
If Φ has no $\mathfrak{M}_*(\Sigma)$-model, then by the previous Henkin Theorem Φ is $\mathfrak{NK}_*$-inconsistent. Since every $\mathfrak{NK}_*$-proof is finite, this means some finite subset Ψ of Φ is $\mathfrak{NK}_*$-inconsistent.
Hence, Ψ has no $\mathfrak{M}_*(\Sigma)$-model.

©Benzmüller & Brown, 2006
UNIVERSITÄT DES SAARLANDES
SEMHOL'06-[5] – p.222

Note on the Saturation Condition

- it may be hard to prove saturation (∇_{sat})

Note on the Saturation Condition

- it may be hard to prove saturation (∇_{sat})
- in fact, as we show in [Unpublished-04] and [IJCAR-06], proving ∇_{sat} is as hard as showing admissibility of cut

Note on the Saturation Condition

- it may be hard to prove saturation (∇_{sat})
- in fact, as we show in [Unpublished-04] and [IJCAR-06], proving ∇_{sat} is as hard as showing admissibility of cut
- if time permits, we will hear more about this later

Model Existence Theorems

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated ACC and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences. For all $* \in \{\ldots\}$ we have:

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated ACC and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences. For all $* \in \{\ldots\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated ACC and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences. For all $* \in \{\ldots\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.
- Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$)

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated ACC and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences. For all $* \in \{\ldots\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.
- Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$)

Proof: The proof combines the following three ingredients:

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated ACC and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences. For all $* \in \{\ldots\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.
- Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$)

Proof: The proof combines the following three ingredients:

Lemma (Compactness of ACC's): For each ACC Γ_Σ there exists a compact ACC Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated ACC and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences. For all $* \in \{\ldots\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.
- Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$)

Proof: The proof combines the following three ingredients:

Lemma (Compactness of ACC's):

Lemma (Abstract Extension Lemma): Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated ACC and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences. For all $* \in \{\ldots\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.
- Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$)

Proof: The proof combines the following three ingredients:

Lemma (Compactness of ACC's):

Lemma (Abstract Extension Lemma):

Thm (Model Existence Theorem for Saturated Hintikka Sets): For all $*$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$. Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_s$.

Model Existence Theorem

Thm.: Let Γ_Σ be a saturated ACC and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of sentences. For all $* \in \{\ldots\}$ we have:

- If Γ_Σ is an $\mathfrak{Acc}_*$, then there exists a $\mathcal{M} \in \mathfrak{M}_*$ that satisfies Φ.
- Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$ (where $\aleph_s$ is the cardinality of $wff_\alpha(\Sigma)$)

Proof: The proof combines the following three ingredients:

Lemma (Compactness of ACC's):

Lemma (Abstract Extension Lemma):

Thm (Model Existence Theorem for Saturated Hintikka Sets):

- ... now we sketch the proofs of these ingredients ...

Compactness od ACC's

Lemma: For each abstract consistency class Γ_Σ there exists a compact abstract consistency class Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Compactness od ACC's

Lemma: For each abstract consistency class Γ_Σ there exists a compact abstract consistency class Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Proof: (following and extending [Andrews-BlackBook])

Compactness od ACC's

Lemma: For each abstract consistency class Γ_Σ there exists a compact abstract consistency class Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Proof: (following and extending [Andrews-BlackBook])

- $\Gamma'_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \text{every finite subset of } \Phi \text{ is in } \Gamma_\Sigma\}$

Compactness od ACC's

Lemma: For each abstract consistency class Γ_Σ there exists a compact abstract consistency class Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Proof: (following and extending [Andrews-BlackBook])

- $\Gamma'_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid \text{every finite subset of } \Phi \text{ is in } \Gamma_\Sigma\}$
- Show $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$: Suppose $\Phi \in \Gamma_\Sigma$. Γ_Σ is closed under subsets, so every finite subset of Φ is in Γ_Σ and thus $\Phi \in \Gamma'_\Sigma$.

Compactness od ACC's

Lemma: For each abstract consistency class Γ_Σ there exists a compact abstract consistency class Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Proof: (following and extending [Andrews-BlackBook])

- $\Gamma'_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid$ every finite subset of Φ is in $\Gamma_\Sigma\}$
- Show $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$:
- Show Γ'_Σ is compact: Suppose $\Phi \in \Gamma'_\Sigma$ and Ψ is an arbitrary finite subset of Φ. By definition of Γ'_Σ all finite subsets of Φ are in Γ_Σ and therefore $\Psi \in \Gamma'_\Sigma$. Thus all finite subsets of Φ are in Γ'_Σ whenever Φ is in Γ'_Σ.
  On the other hand, suppose all finite subsets of Φ are in Γ'_Σ. Then by the definition of Γ'_Σ the finite subsets of Φ are also in Γ_Σ, so $\Phi \in \Gamma'_\Sigma$.
  Thus Γ'_Σ is compact (and closed under subsets).

Compactness od ACC's

Lemma: For each abstract consistency class Γ_Σ there exists a compact abstract consistency class Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Proof: (following and extending [Andrews-BlackBook])

- $\Gamma'_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid$ every finite subset of Φ is in $\Gamma_\Sigma\}$
- Show $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$:
- Show Γ'_Σ is compact:
- Show that Γ'_Σ satisfies ∇_* whenever Γ_Σ satisfies ∇_*:

©Benzmüller & Brown, 2006 SEMHOL'06-[6] – p.226

Compactness od ACC's

Lemma: For each abstract consistency class Γ_Σ there exists a compact abstract consistency class Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Proof: (following and extending [Andrews-BlackBook])

- $\Gamma'_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid$ every finite subset of Φ is in $\Gamma_\Sigma\}$
- Show $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$:
- Show Γ'_Σ is compact:
- Show that Γ'_Σ satisfies ∇_* whenever Γ_Σ satisfies ∇_*:

∇_c Let $\Phi \in \Gamma'_\Sigma$ and suppose there is an atom $\mathbf{A}$, such that $\{\mathbf{A}, \neg\mathbf{A}\} \subseteq \Phi$. $\{\mathbf{A}, \neg\mathbf{A}\}$ is clearly a finite subset of Φ and hence $\{\mathbf{A}, \neg\mathbf{A}\} \in \Gamma_\Sigma$ contradicting ∇_c for Γ_Σ.

©Benzmüller & Brown, 2006 SEMHOL'06-[6] – p.226

Compactness od ACC's

Lemma: For each abstract consistency class Γ_Σ there exists a compact abstract consistency class Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Proof: (following and extending [Andrews-BlackBook])

- $\Gamma'_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid$ every finite subset of Φ is in $\Gamma_\Sigma\}$
- Show $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$:
- Show Γ'_Σ is compact:
- Show that Γ'_Σ satisfies ∇_* whenever Γ_Σ satisfies ∇_*:

$\nabla_\neg$ Let $\Phi \in \Gamma'_\Sigma$, $\neg\neg\mathbf{A} \in \Phi$, Ψ be any finite subset of $\Phi * \mathbf{A}$, and $\Theta := (\Psi \setminus \{\mathbf{A}\}) * \neg\neg\mathbf{A}$. Θ is a finite subset of Φ, so $\Theta \in \Gamma_\Sigma$. Since Γ_Σ is an abstract consistency class and $\neg\neg\mathbf{A} \in \Theta$, we get $\Theta * \mathbf{A} \in \Gamma_\Sigma$ by $\nabla_\neg$ for Γ_Σ. We know that $\Psi \subseteq \Theta * \mathbf{A}$ and Γ_Σ is closed under subsets, so $\Psi \in \Gamma_\Sigma$. Thus every finite subset Ψ of $\Phi * \mathbf{A}$ is in Γ_Σ and therefore by definition $\Phi * \mathbf{A} \in \Gamma'_\Sigma$.

©Benzmüller & Brown, 2006 SEMHOL'06-[6] – p.226

Compactness od ACC's

Lemma: For each abstract consistency class Γ_Σ there exists a compact abstract consistency class Γ'_Σ satisfying the same ∇_* properties such that $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Proof: (following and extending [Andrews-BlackBook])

- $\Gamma'_\Sigma := \{\Phi \subseteq cwff_o(\Sigma) \mid$ every finite subset of Φ is in $\Gamma_\Sigma\}$
- Show $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$:
- Show Γ'_Σ is compact:
- Show that Γ'_Σ satisfies ∇_* whenever Γ_Σ satisfies ∇_*:

  For $\nabla_\beta, \nabla_\eta, \nabla_\vee, \nabla_\wedge, \nabla_\forall, \nabla_\exists, \nabla_\xi, \nabla_\mathfrak{f}, \nabla_\mathfrak{b}, \nabla_{sat}$ see the lecture notes.

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n \geq 0} \mathcal{H}^n$.

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n \geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n \geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n \geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.
  - If $\mathcal{H}^n * \mathbf{A}^n \in \Gamma_\Sigma$, then $\mathcal{H}^{n+1} := \mathcal{H}^n * \mathbf{A}^n * \mathbf{E}^n * \mathbf{X}^n$, where

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n \geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.
  - If $\mathcal{H}^n * \mathbf{A}^n \in \Gamma_\Sigma$, then $\mathcal{H}^{n+1} := \mathcal{H}^n * \mathbf{A}^n * \mathbf{E}^n * \mathbf{X}^n$, where

$\mathbf{E}^n$: $\mathbf{E}^n := \neg(\mathbf{B}w^n_\alpha)$ if $\mathbf{A}^n$ is of the orm $\neg(\Pi^\alpha\mathbf{B})$, and let
$\mathbf{E}^n := \mathbf{A}^n$ otherwise

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n\geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.
  - If $\mathcal{H}^n * \mathbf{A}^n \in \Gamma_\Sigma$, then $\mathcal{H}^{n+1} := \mathcal{H}^n * \mathbf{A}^n * \mathbf{E}^n * \mathbf{X}^n$, where

$\mathbf{X}^n$: If $* \in \{\beta\mathfrak{f}, \beta\mathfrak{fb}\}$ and $\mathbf{A}^n$ is of the form $\neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G})$, let $\mathbf{X}^n := \neg(\mathbf{F}w_\alpha^n \doteq^\beta \mathbf{G}w_\alpha^n)$.
If $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$ and $\mathbf{A}^n$ is of the form $\neg((\lambda X_\alpha.\mathbf{M}) \doteq^{\alpha\to\beta} (\lambda X.\mathbf{N}))$, let $\mathbf{X}^n := \neg([w_\alpha^n/X]\mathbf{M} \doteq^\beta [w_\alpha^n/X]\mathbf{N})$. Otherwise, let $\mathbf{X}^n := \mathbf{A}^n$.

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n\geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.
  - If $\mathcal{H}^n * \mathbf{A}^n \in \Gamma_\Sigma$, then $\mathcal{H}^{n+1} := \mathcal{H}^n * \mathbf{A}^n * \mathbf{E}^n * \mathbf{X}^n$, where

params w_α^n: need to prove that always fresh parameters exists

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n\geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.
  - If $\mathcal{H}^n * \mathbf{A}^n \in \Gamma_\Sigma$, then $\mathcal{H}^{n+1} := \mathcal{H}^n * \mathbf{A}^n * \mathbf{E}^n * \mathbf{X}^n$, where

generalize: the above only works for the countable case; in the lecture notes we use transfinite induction for the general case

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n\geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.
  - If $\mathcal{H}^n * \mathbf{A}^n \in \Gamma_\Sigma$, then $\mathcal{H}^{n+1} := \mathcal{H}^n * \mathbf{A}^n * \mathbf{E}^n * \mathbf{X}^n$, where
  - Then we show by induction that $\mathcal{H}^n \in \Gamma_\Sigma$ for all n.

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n \geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.
  - If $\mathcal{H}^n * \mathbf{A}^n \in \Gamma_\Sigma$, then $\mathcal{H}^{n+1} := \mathcal{H}^n * \mathbf{A}^n * \mathbf{E}^n * \mathbf{X}^n$, where
  - Then we show by induction that $\mathcal{H}^n \in \Gamma_\Sigma$ for all n.
  - Since Γ_Σ is compact, we also have $\mathcal{H} \in \Gamma_\Sigma$.

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n \geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.
  - If $\mathcal{H}^n * \mathbf{A}^n \in \Gamma_\Sigma$, then $\mathcal{H}^{n+1} := \mathcal{H}^n * \mathbf{A}^n * \mathbf{E}^n * \mathbf{X}^n$, where
  - Then we show by induction that $\mathcal{H}^n \in \Gamma_\Sigma$ for all n.
  - Since Γ_Σ is compact, we also have $\mathcal{H} \in \Gamma_\Sigma$.
  - Hence $\Phi \subseteq \mathcal{H}$ and $\mathcal{H} \in \Gamma_\Sigma$.

Abstract Extension Lemma

Lemma: Let Σ be a signature, Γ_Σ be a compact ACC in $\mathfrak{Acc}_*$, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, and let $\Phi \in \Gamma_\Sigma$ be sufficiently pure. Then there exists a Σ-Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$, such that $\Phi \subseteq \mathcal{H}$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ is saturated.

Proof: (We only give the simplified idea; see lecture notes for details)

- Construct $\mathcal{H}$ as follows: $\mathcal{H} := \bigcup_{n \geq 0} \mathcal{H}^n$.
  - $\mathcal{H}^0 := \Phi$.
  - If $\mathcal{H}^n * \mathbf{A}^n \notin \Gamma_\Sigma$, we let $\mathcal{H}^{n+1} := \mathcal{H}^n$.
  - If $\mathcal{H}^n * \mathbf{A}^n \in \Gamma_\Sigma$, then $\mathcal{H}^{n+1} := \mathcal{H}^n * \mathbf{A}^n * \mathbf{E}^n * \mathbf{X}^n$, where
  - Then we show by induction that $\mathcal{H}^n \in \Gamma_\Sigma$ for all n.
  - Since Γ_Σ is compact, we also have $\mathcal{H} \in \Gamma_\Sigma$.
  - Hence $\Phi \subseteq \mathcal{H}$ and $\mathcal{H} \in \Gamma_\Sigma$.
  - Remains to show that $\mathcal{H}$ is (subset) maximal in Γ_Σ and that $\mathcal{H}$ is indeed a Hintikka set.

Hintikka Sets

- Hintikka sets connect syntax with semantics as they provide the basis for the model constructions in the model existence theorem(s).

Hintikka Sets

- Hintikka sets connect syntax with semantics as they provide the basis for the model constructions in the model existence theorem(s).
- We have defined eight different notions of abstract consistency classes by first defining properties ∇_*, then specifying which should hold in $\mathfrak{Acc}_*$.

Hintikka Sets

- Hintikka sets connect syntax with semantics as they provide the basis for the model constructions in the model existence theorem(s).
- We have defined eight different notions of abstract consistency classes by first defining properties ∇_*, then specifying which should hold in $\mathfrak{Acc}_*$.
- Similarly, we define Hintikka sets by first defining the desired properties.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{C}, \mathbf{D} \in cwff_\alpha(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha.\mathbf{M}), (\lambda X.\mathbf{N}), \mathbf{G}, \mathbf{H} \in cwff_{\alpha\to\beta}(\Sigma)$:

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{C}, \mathbf{D} \in cwff_\alpha(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha.\mathbf{M}), (\lambda X.\mathbf{N}), \mathbf{G}, \mathbf{H} \in cwff_{\alpha\to\beta}(\Sigma)$:

$\vec{\nabla}_c$ $\mathbf{A} \notin \mathcal{H}$ or $\neg\mathbf{A} \notin \mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_{o}(\Sigma)$, $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_{\alpha}(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha \to o}(\Sigma)$, and $(\lambda X_{\alpha} \mathbf{.M}), (\lambda X \mathbf{.N}), \mathbf{G}, \mathbf{H} \in \mathit{cwff}_{\alpha \to \beta}(\Sigma)$:

$\vec{\nabla}_c$ $\mathbf{A} \notin \mathcal{H}$ or $\neg\mathbf{A} \notin \mathcal{H}$.

$\vec{\nabla}_\neg$ If $\neg\neg\mathbf{A} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_{o}(\Sigma)$, $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_{\alpha}(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha \to o}(\Sigma)$, and $(\lambda X_{\alpha} \mathbf{.M}), (\lambda X \mathbf{.N}), \mathbf{G}, \mathbf{H} \in \mathit{cwff}_{\alpha \to \beta}(\Sigma)$:

$\vec{\nabla}_c$ $\mathbf{A} \notin \mathcal{H}$ or $\neg\mathbf{A} \notin \mathcal{H}$.

$\vec{\nabla}_\neg$ If $\neg\neg\mathbf{A} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$.

$\vec{\nabla}_\beta$ If $\mathbf{A} \in \mathcal{H}$ and $\mathbf{A} =_\beta \mathbf{B}$, then $\mathbf{B} \in \mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_{o}(\Sigma)$, $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_{\alpha}(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha \to o}(\Sigma)$, and $(\lambda X_{\alpha} \mathbf{.M}), (\lambda X \mathbf{.N}), \mathbf{G}, \mathbf{H} \in \mathit{cwff}_{\alpha \to \beta}(\Sigma)$:

$\vec{\nabla}_c$ $\mathbf{A} \notin \mathcal{H}$ or $\neg\mathbf{A} \notin \mathcal{H}$.

$\vec{\nabla}_\neg$ If $\neg\neg\mathbf{A} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$.

$\vec{\nabla}_\beta$ If $\mathbf{A} \in \mathcal{H}$ and $\mathbf{A} =_\beta \mathbf{B}$, then $\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$ or $\mathbf{B} \in \mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_{o}(\Sigma)$, $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_{\alpha}(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha \to o}(\Sigma)$, and $(\lambda X_{\alpha} \mathbf{.M}), (\lambda X \mathbf{.N}), \mathbf{G}, \mathbf{H} \in \mathit{cwff}_{\alpha \to \beta}(\Sigma)$:

$\vec{\nabla}_c$ $\mathbf{A} \notin \mathcal{H}$ or $\neg\mathbf{A} \notin \mathcal{H}$.

$\vec{\nabla}_\neg$ If $\neg\neg\mathbf{A} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$.

$\vec{\nabla}_\beta$ If $\mathbf{A} \in \mathcal{H}$ and $\mathbf{A} =_\beta \mathbf{B}$, then $\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$ or $\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \mathcal{H}$, then $\neg\mathbf{A} \in \mathcal{H}$ and $\neg\mathbf{B} \in \mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_\alpha(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha \mathbf{.M}), (\lambda X \mathbf{.N}), \mathbf{G}, \mathbf{H} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$:

$\vec{\nabla}_c$ $\mathbf{A} \notin \mathcal{H}$ or $\neg\mathbf{A} \notin \mathcal{H}$.

$\vec{\nabla}_\neg$ If $\neg\neg\mathbf{A} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$.

$\vec{\nabla}_\beta$ If $\mathbf{A} \in \mathcal{H}$ and $\mathbf{A} =_\beta \mathbf{B}$, then $\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$ or $\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \mathcal{H}$, then $\neg\mathbf{A} \in \mathcal{H}$ and $\neg\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\forall$ If $\Pi^\alpha\mathbf{F} \in \mathcal{H}$, then $\mathbf{FW} \in \mathcal{H}$ for each $\mathbf{W} \in \mathit{cwff}_\alpha(\Sigma)$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_\alpha(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha \mathbf{.M}), (\lambda X \mathbf{.N}), \mathbf{G}, \mathbf{H} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$:

$\vec{\nabla}_c$ $\mathbf{A} \notin \mathcal{H}$ or $\neg\mathbf{A} \notin \mathcal{H}$.

$\vec{\nabla}_\neg$ If $\neg\neg\mathbf{A} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$.

$\vec{\nabla}_\beta$ If $\mathbf{A} \in \mathcal{H}$ and $\mathbf{A} =_\beta \mathbf{B}$, then $\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\vee$ If $\mathbf{A} \vee \mathbf{B} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$ or $\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\wedge$ If $\neg(\mathbf{A} \vee \mathbf{B}) \in \mathcal{H}$, then $\neg\mathbf{A} \in \mathcal{H}$ and $\neg\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\forall$ If $\Pi^\alpha\mathbf{F} \in \mathcal{H}$, then $\mathbf{FW} \in \mathcal{H}$ for each $\mathbf{W} \in \mathit{cwff}_\alpha(\Sigma)$.

$\vec{\nabla}_\exists$ If $\neg\Pi^\alpha\mathbf{F} \in \mathcal{H}$, then there is a parameter $w_\alpha \in \Sigma_\alpha$ such that $\neg(\mathbf{F}w) \in \mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_\alpha(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha \mathbf{.M}), (\lambda X \mathbf{.N}), \mathbf{G}, \mathbf{H} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$:

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_\alpha(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha \mathbf{.M}), (\lambda X \mathbf{.N}), \mathbf{G}, \mathbf{H} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$:

$\vec{\nabla}_\mathfrak{b}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \mathcal{H}$, then $\{\mathbf{A}, \neg\mathbf{B}\} \subseteq \mathcal{H}$ or $\{\neg\mathbf{A}, \mathbf{B}\} \subseteq \mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A},\mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{C},\mathbf{D} \in \mathit{cwff}_\alpha(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha.\mathbf{M}), (\lambda X.\mathbf{N}), \mathbf{G},\mathbf{H} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$:

$\vec\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \mathcal{H}$, then $\{\mathbf{A},\neg\mathbf{B}\} \subseteq \mathcal{H}$ or $\{\neg\mathbf{A},\mathbf{B}\}\subseteq\mathcal{H}$.

$\vec\nabla_\eta$ If $\mathbf{A}\in\mathcal{H}$ and $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$, then $\mathbf{B}\in\mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A},\mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{C},\mathbf{D} \in \mathit{cwff}_\alpha(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha.\mathbf{M}), (\lambda X.\mathbf{N}), \mathbf{G},\mathbf{H} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$:

$\vec\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \mathcal{H}$, then $\{\mathbf{A},\neg\mathbf{B}\} \subseteq \mathcal{H}$ or $\{\neg\mathbf{A},\mathbf{B}\}\subseteq\mathcal{H}$.

$\vec\nabla_\eta$ If $\mathbf{A}\in\mathcal{H}$ and $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$, then $\mathbf{B}\in\mathcal{H}$.

$\vec\nabla_\xi$ If $\neg(\lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \mathcal{H}$, then there is a parameter $w_\alpha \in \Sigma_\alpha$ such that $\neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A},\mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{C},\mathbf{D} \in \mathit{cwff}_\alpha(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha.\mathbf{M}), (\lambda X.\mathbf{N}), \mathbf{G},\mathbf{H} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$:

$\vec\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \mathcal{H}$, then $\{\mathbf{A},\neg\mathbf{B}\} \subseteq \mathcal{H}$ or $\{\neg\mathbf{A},\mathbf{B}\}\subseteq\mathcal{H}$.

$\vec\nabla_\eta$ If $\mathbf{A}\in\mathcal{H}$ and $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$, then $\mathbf{B}\in\mathcal{H}$.

$\vec\nabla_\xi$ If $\neg(\lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \mathcal{H}$, then there is a parameter $w_\alpha \in \Sigma_\alpha$ such that $\neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \mathcal{H}$.

$\vec\nabla_{\mathfrak{f}}$ If $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H})\in\mathcal{H}$, then there is a parameter $w_\alpha\in\Sigma_\alpha$ such that $\neg(\mathbf{G}w \doteq^\beta \mathbf{H}w)\in\mathcal{H}$.

Hintikka Properties

Defn.: Let $\mathcal{H}$ be a set of sentences. We define the following properties which $\mathcal{H}$ may satisfy, where $\mathbf{A},\mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{C},\mathbf{D} \in \mathit{cwff}_\alpha(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$, and $(\lambda X_\alpha.\mathbf{M}), (\lambda X.\mathbf{N}), \mathbf{G},\mathbf{H} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$:

$\vec\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \mathcal{H}$, then $\{\mathbf{A},\neg\mathbf{B}\} \subseteq \mathcal{H}$ or $\{\neg\mathbf{A},\mathbf{B}\}\subseteq\mathcal{H}$.

$\vec\nabla_\eta$ If $\mathbf{A}\in\mathcal{H}$ and $\mathbf{A} \stackrel{\beta\eta}{=} \mathbf{B}$, then $\mathbf{B}\in\mathcal{H}$.

$\vec\nabla_\xi$ If $\neg(\lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \mathcal{H}$, then there is a parameter $w_\alpha \in \Sigma_\alpha$ such that $\neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N}) \in \mathcal{H}$.

$\vec\nabla_{\mathfrak{f}}$ If $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H})\in\mathcal{H}$, then there is a parameter $w_\alpha\in\Sigma_\alpha$ such that $\neg(\mathbf{G}w \doteq^\beta \mathbf{H}w)\in\mathcal{H}$.

$\vec\nabla_{sat}$ Either $\mathbf{A}\in\mathcal{H}$ or $\neg\mathbf{A}\in\mathcal{H}$.

Σ-Hintikka Set

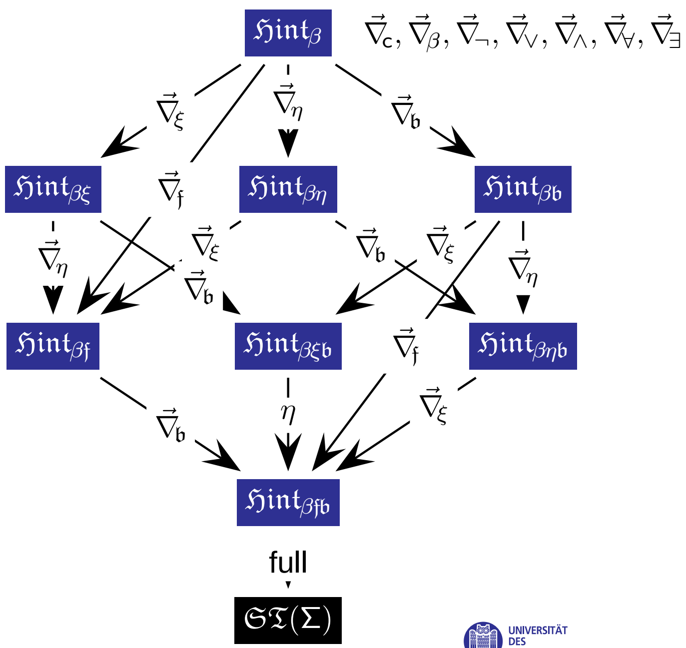
Defn.: A set $\mathcal{H}$ of sentences is called a Σ-Hintikka set if it satisfies $\vec{\nabla}_c$, $\vec{\nabla}_\neg$, $\vec{\nabla}_\beta$, $\vec{\nabla}_\vee$, $\vec{\nabla}_\wedge$, $\vec{\nabla}_\forall$ and $\vec{\nabla}_\exists$.

Σ-Hintikka Set

Defn.: A set $\mathcal{H}$ of sentences is called a Σ-Hintikka set if it satisfies $\vec{\nabla}_c$, $\vec{\nabla}_\neg$, $\vec{\nabla}_\beta$, $\vec{\nabla}_\vee$, $\vec{\nabla}_\wedge$, $\vec{\nabla}_\forall$ and $\vec{\nabla}_\exists$.

- We define the following collections of Hintikka sets: $\mathfrak{Hint}_\beta$, $\mathfrak{Hint}_{\beta\eta}$, $\mathfrak{Hint}_{\beta\xi}$, $\mathfrak{Hint}_{\beta\mathfrak{f}}$, $\mathfrak{Hint}_{\beta\mathfrak{b}}$, $\mathfrak{Hint}_{\beta\eta\mathfrak{b}}$, $\mathfrak{Hint}_{\beta\xi\mathfrak{b}}$, and $\mathfrak{Hint}_{\beta\mathfrak{fb}}$, where we indicate by indices which additional properties from $\{\vec{\nabla}_\eta, \vec{\nabla}_\xi, \vec{\nabla}_\mathfrak{f}, \vec{\nabla}_\mathfrak{b}\}$ are required.

Σ-Hintikka Sets



Model Ex. Thm for Saturated H.-Sets

Thm.: (Model Existence Theorem for Saturated Hintikka Sets) For all $* \in \{\ldots\}$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$. Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_s$.

Model Ex. Thm for Saturated H.-Sets

Thm.: (Model Existence Theorem for Saturated Hintikka Sets)
For all $* \in \{\ldots\}$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$.
Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_\mathsf{s}$.

Proof: (we only sketch the idea)

Model Ex. Thm for Saturated H.-Sets

Thm.: (Model Existence Theorem for Saturated Hintikka Sets)
For all $* \in \{\ldots\}$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$.
Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_\mathsf{s}$.

Proof: (we only sketch the idea)

- We construct Σ-model $\mathcal{M}_1^{\mathcal{H}} := (\mathit{cwff}(\Sigma){\downarrow_\beta}, @^\beta, \mathcal{E}^\beta, \upsilon)$ for $\mathcal{H}$

Model Ex. Thm for Saturated H.-Sets

Thm.: (Model Existence Theorem for Saturated Hintikka Sets)
For all $* \in \{\ldots\}$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$.
Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_\mathsf{s}$.

Proof: (we only sketch the idea)

- We construct Σ-model $\mathcal{M}_1^{\mathcal{H}} := (\mathit{cwff}(\Sigma){\downarrow_\beta}, @^\beta, \mathcal{E}^\beta, \upsilon)$ for $\mathcal{H}$
- $\mathcal{M}_1^{\mathcal{H}}$ is based on term evaluation $\mathcal{TE}(\Sigma)^\beta := (\mathit{cwff}(\Sigma){\downarrow_\beta}, @^\beta, \mathcal{E}^\beta)$ where
  - $\mathit{cwff}(\Sigma){\downarrow_\beta}$: closed well-formed formulae in β-normal form
  - $\mathbf{A} @^\beta \mathbf{B} := (\mathbf{AB}){\downarrow_\beta}$
  - $\mathcal{E}_\varphi^\beta(\mathbf{A}) := \sigma(\mathbf{A}){\downarrow_\beta}$

Model Ex. Thm for Saturated H.-Sets

Thm.: (Model Existence Theorem for Saturated Hintikka Sets)
For all $* \in \{\ldots\}$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$.
Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_\mathsf{s}$.

Proof: (we only sketch the idea)

- We construct Σ-model $\mathcal{M}_1^{\mathcal{H}} := (\mathit{cwff}(\Sigma){\downarrow_\beta}, @^\beta, \mathcal{E}^\beta, \upsilon)$ for $\mathcal{H}$
- $\upsilon(\mathbf{A}) := \begin{cases} \mathtt{T} & \text{if } \mathbf{A} \in \mathcal{H} \\ \mathtt{F} & \text{if } \mathbf{A} \notin \mathcal{H} \end{cases}$

Model Ex. Thm for Saturated H.-Sets

Thm.: (Model Existence Theorem for Saturated Hintikka Sets)
For all $* \in \{\ldots\}$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$. Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_s$.

Proof: (we only sketch the idea)

- We construct Σ-model $\mathcal{M}_1^{\mathcal{H}} := (\mathit{cwff}(\Sigma)\!\downarrow_\beta, @^\beta, \mathcal{E}^\beta, \upsilon)$ for $\mathcal{H}$
- $\mathcal{M}_1^{\mathcal{H}} \models \mathcal{H}$ since $\upsilon(\mathbf{A}) = \mathtt{T}$ for every $\mathbf{A} \in \mathcal{H}$ by definition

Model Ex. Thm for Saturated H.-Sets

Thm.: (Model Existence Theorem for Saturated Hintikka Sets)
For all $* \in \{\ldots\}$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$. Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_s$.

Proof: (we only sketch the idea)

- We construct Σ-model $\mathcal{M}_1^{\mathcal{H}} := (\mathit{cwff}(\Sigma)\!\downarrow_\beta, @^\beta, \mathcal{E}^\beta, \upsilon)$ for $\mathcal{H}$
- $\mathcal{M}_1^{\mathcal{H}} \models \mathcal{H}$ since $\upsilon(\mathbf{A}) = \mathtt{T}$ for every $\mathbf{A} \in \mathcal{H}$ by definition
- may hold: $\mathcal{M}_1^{\mathcal{H}} \notin \mathfrak{M}_*$ as it may not satisfy property $\mathfrak{q}$

Model Ex. Thm for Saturated H.-Sets

Thm.: (Model Existence Theorem for Saturated Hintikka Sets)
For all $* \in \{\ldots\}$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$. Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_s$.

Proof: (we only sketch the idea)

- We construct Σ-model $\mathcal{M}_1^{\mathcal{H}} := (\mathit{cwff}(\Sigma)\!\downarrow_\beta, @^\beta, \mathcal{E}^\beta, \upsilon)$ for $\mathcal{H}$
- $\mathcal{M}_1^{\mathcal{H}} \models \mathcal{H}$ since $\upsilon(\mathbf{A}) = \mathtt{T}$ for every $\mathbf{A} \in \mathcal{H}$ by definition
- may hold: $\mathcal{M}_1^{\mathcal{H}} \notin \mathfrak{M}_*$ as it may not satisfy property $\mathfrak{q}$
- way out: use congruence relation $\dot\sim$ on $\mathcal{M}_1^{\mathcal{H}}$

$$\mathbf{A}_\alpha \dot\sim \mathbf{B}_\alpha \text{ iff } \upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq \mathbf{B})) = \mathtt{T}$$

to construct $\mathcal{M} := \mathcal{M}_1^{\mathcal{H}}/_{\dot\sim}$

Model Ex. Thm for Saturated H.-Sets

Thm.: (Model Existence Theorem for Saturated Hintikka Sets)
For all $* \in \{\ldots\}$ we have: If $\mathcal{H}$ is a saturated Hintikka set in $\mathfrak{Hint}_*$, then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ that satisfies $\mathcal{H}$. Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_s$.

Proof: (we only sketch the idea)

- We construct Σ-model $\mathcal{M}_1^{\mathcal{H}} := (\mathit{cwff}(\Sigma)\!\downarrow_\beta, @^\beta, \mathcal{E}^\beta, \upsilon)$ for $\mathcal{H}$
- $\mathcal{M}_1^{\mathcal{H}} \models \mathcal{H}$ since $\upsilon(\mathbf{A}) = \mathtt{T}$ for every $\mathbf{A} \in \mathcal{H}$ by definition
- may hold: $\mathcal{M}_1^{\mathcal{H}} \notin \mathfrak{M}_*$ as it may not satisfy property $\mathfrak{q}$
- way out: use congruence relation $\dot\sim$ on $\mathcal{M}_1^{\mathcal{H}}$

$$\mathbf{A}_\alpha \dot\sim \mathbf{B}_\alpha \text{ iff } \upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq \mathbf{B})) = \mathtt{T}$$

to construct $\mathcal{M} := \mathcal{M}_1^{\mathcal{H}}/_{\dot\sim}$
- then show that $\mathcal{M}$ 'does the job'

Further Reading

- [Chris-PhD-99] C. Benzmüller: Equality and Extensionality in Higher-Order Theorem Proving. Doctoral Thesis, Computer Science, Saarland University, 1999.

Further Reading

- [Chris-PhD-99] C. Benzmüller: Equality and Extensionality in Higher-Order Theorem Proving. Doctoral Thesis, Computer Science, Saarland University, 1999.
- [Chad-PhD-04] C. Brown: Set Comprehension in Church's Type Theory, Ph.D. Dissertation, Mathematics Department, Carnegie Mellon University, 2004)

Further Reading

- [Chris-PhD-99] C. Benzmüller: Equality and Extensionality in Higher-Order Theorem Proving. Doctoral Thesis, Computer Science, Saarland University, 1999.
- [Chad-PhD-04] C. Brown: Set Comprehension in Church's Type Theory, Ph.D. Dissertation, Mathematics Department, Carnegie Mellon University, 2004)
- [JSL-04] C. Benzmüller, C. Brown, and M. Kohlhase: Higher Order Semantics and Extensionality. Journal of Symbolic Logic. (2004) 69(4):1027-1088. ⓒJSTOR.

Further Reading

- [Chris-PhD-99] C. Benzmüller: Equality and Extensionality in Higher-Order Theorem Proving. Doctoral Thesis, Computer Science, Saarland University, 1999.
- [Chad-PhD-04] C. Brown: Set Comprehension in Church's Type Theory, Ph.D. Dissertation, Mathematics Department, Carnegie Mellon University, 2004)
- [JSL-04] C. Benzmüller, C. Brown, and M. Kohlhase: Higher Order Semantics and Extensionality. Journal of Symbolic Logic. (2004) 69(4):1027-1088. ⓒJSTOR.
- [Unpublished-04] C. Benzmüller, C. E. Brown, M. Kohlhase: Semantic Techniques for Higher-Order Cut-Elimination. Seki-Report SR-04-07 (ISSN 1437-4447), Saarland University, 2004.

Further Reading

- [Chris-PhD-99] C. Benzmüller: Equality and Extensionality in Higher-Order Theorem Proving. Doctoral Thesis, Computer Science, Saarland University, 1999.
- [Chad-PhD-04] C. Brown: Set Comprehension in Church's Type Theory, Ph.D. Dissertation, Mathematics Department, Carnegie Mellon University, 2004)
- [JSL-04] C. Benzmüller, C. Brown, and M. Kohlhase: Higher Order Semantics and Extensionality. Journal of Symbolic Logic. (2004) 69(4):1027-1088. ©JSTOR.
- [Unpublished-04] C. Benzmüller, C. E. Brown, M. Kohlhase: Semantic Techniques for Higher-Order Cut-Elimination. Seki-Report SR-04-07 (ISSN 1437-4447), Saarland University, 2004.
- [TPHOLs-05] C. Benzmüller, C. Brown: A Structured Set of Higher-Order Problems. TPHOLs 2005: 66-81. ©Springer.

Further Reading

- [Chris-PhD-99] C. Benzmüller: Equality and Extensionality in Higher-Order Theorem Proving. Doctoral Thesis, Computer Science, Saarland University, 1999.
- [Chad-PhD-04] C. Brown: Set Comprehension in Church's Type Theory, Ph.D. Dissertation, Mathematics Department, Carnegie Mellon University, 2004)
- [JSL-04] C. Benzmüller, C. Brown, and M. Kohlhase: Higher Order Semantics and Extensionality. Journal of Symbolic Logic. (2004) 69(4):1027-1088. ©JSTOR.
- [Unpublished-04] C. Benzmüller, C. E. Brown, M. Kohlhase: Semantic Techniques for Higher-Order Cut-Elimination. Seki-Report SR-04-07 (ISSN 1437-4447), Saarland University, 2004.
- [TPHOLs-05] C. Benzmüller, C. Brown: A Structured Set of Higher-Order Problems. TPHOLs 2005: 66-81. ©Springer.

- [IJCAR-06] C. Benzmüller, C. Brown, and M. Kohlhase: Cut-Simulation in Impredicative Logics. Third International Joint Conference on Automated Reasoning (IJCAR'06), LNAI , Seattle, USA, 2006. ©Springer. To appear.

  (see also: C. Benzmüller, C. E. Brown, M. Kohlhase: Cut-Simulation in Impredicative Logics (Extended Version). Seki-Report SR-2006-01 (ISSN 1437-4447), Saarland University, 2004.)

Thank You!