Semantic Techniques for Cut Elimination in Higher Order Logics

Christoph Benzmüller[1], Chad E. Brown[2], and Michael Kohlhase[1,2]
[1]Saarland University, Germany
[2]Carnegie Mellon University, Pittsburgh, USA
chris|kohlhase@cs.uni-sb.de, cebrown@andrew.cmu.edu

Draft of January 13, 2003
Do not distribute

Abstract

In this report we present a variety of model existence theorems for classical higher order logics. They extend the well-known abstract consistency methods to higher order logics and pinpoint the contribution of extensionality.

The results reported here weaken the saturation precondition needed in earlier model existence theorems, making them appliccable to cut-free higher order logics.

This report reprints the article "Higher Order Semantics and Extensionality" under review by the Journal of Symbolic Logic and develops the cut-free extensions in detail.

Contents

Chapter 1

Higher Order Semantics and Extensionality

In this chapter we reprint the CMU Techreport CMU-01-03 [BBK03] for convenience.

In that report we re-examine the semantics of classical higher order logic with the purpose of clarifying the role of extensionality. To reach this goal, we distinguish nine classes of higher order models with respect to various combinations of Boolean extensionality and three forms of functional extensionality. Furthermore, we develop a methodology of abstract consistency methods (by providing the necessary model existence theorems) needed to analyze completeness of (machine-oriented) higher order calculi with respect to these model classes.

1.1 Motivation

In classical first order predicate logic, it is rather simple to assess the deductive power of a calculus: first order logic has a well-established and intuitive set-theoretic semantics, relative to which completeness can easily be verified using, for instance, the abstract consistency method (cf. the introductory textbooks [And02, Fit96]) This well-understood meta-theory has supported the development of calculi adapted to special applications – such as automated theorem proving (cf. [BS98, RV01] for an overview).

In higher order logics, the situation is rather different: the intuitive set-theoretic standard semantics cannot give a sensible notion of completeness, since it does not admit complete (recursively axiomatizable) calculi [Göd31, And02]. There is a more general notion of semantics [Hen50], the so-called Henkin-models, that allows complete (recursively axiomatizable) calculi and therefore sets the standard for deductive power of calculi.

Peter Andrews' *Unifying Principle for Type Theory* [And71] provides a method of higher order abstract consistency that has become the standard tool for completeness proofs in higher order logic, even though it can only be used to show completeness relative to a certain Hilbert style calculus $\mathfrak{T}_\beta$. A calculus $\mathcal{C}$ is called complete relative to a calculus $\mathfrak{T}_\beta$ iff (if and only if) $\mathcal{C}$ proves all theorems of $\mathfrak{T}_\beta$. Since $\mathfrak{T}_\beta$ is not complete with respect to Henkin models, the notion of completeness that can be established by this method is a strictly weaker notion than Henkin completeness. The difference between these notions of completeness can largely be analyzed in terms of availability of various extensionality principles, which can be expressed axiomatically in higher order logic.

As a consequence of the limitations of Andrew's *Unifying Principle*, calculi for higher order automated theorem proving [And71, Hue72, Hue73, JP72, Mil83, Koh94, Koh95] and the corresponding theorem proving systems such as TPS [ABB00, ABI$^+$96], or earlier versions of the LEO system are not or cannot be proven complete with respect to Henkin models. Moreover, they are not even sound with respect to $\mathfrak{T}_\beta$, since all of them but the first employ η-conversion, which is not admissible in $\mathfrak{T}_\beta$. In other words, their deductive power lies somewhere between $\mathfrak{T}_\beta$ and Henkin models. Characterizing exactly where reveals important theoretical properties of these calculi that have direct consequences for the adequacy in various application domains (see the discussion below in section 1.8.1). Unlike calculi without computational concerns, calculi for mechanized reasoning systems cannot be made complete by simply adding extensionality axioms, since the search spaces induced by their introduction grow prohibitively. Being able to compare and characterize the methods and computational devices used instead is a prerequisite for further development in this area.

In this situation, the aim of this article is to provide a semantical meta theory that will support the development of higher order calculi for automated theorem proving just as the corresponding methodology does in first order logic. To reach this goal, we need to establish.

1. classes of models that adequately characterize the deductive power of existing theorem-proving calculi (providing semantics with respect to which they are sound and complete), and

2. a methodology of abstract consistency methods (by providing for these model classes the necessary model existence theorems, which extend Andrews' Unifying Principle), so that the completeness analysis for higher order calculi will become almost as simple as in first order logic.

We fully achieve the first goal in this article, and take a large step towards the second. In the model existence theorems presented in this article, we have to assume a new condition called *saturation*, which limits their utility in completeness proofs for machine-oriented calculi. Fortunately, the saturation condition can be lifted by extensions of the methods presented in this article (see the discussion in the conclusion 1.8.2 and [BBK02b]).

Due to the inherent complexity of higher order semantics we first give an informal exposition of the issues covered and the techniques applied. In Section 1.4, we will investigate the properties of the model classes introduced in Section 1.3 in more detail and corroborate them with example models in Section 1.5. We prove model existence theorems for the model classes in Section 1.6. Finally, in Section 1.7 we will apply the model existence theorems from Section 1.6 to the task of proving completeness of higher order natural deduction calculi. Section 1.8 concludes the article with a discussion of related work, possible applications, and the saturation assumption we introduced for the model existence theorems.

The work reported in this article is based on [BK97] and significantly extends the material presented there.

1.2 Informal Exposition

Before we turn to the exposition of the semantics in Section 1.2.3, let us specify what we mean by "higher order logic": any simply typed logical system that allows quantification over function and predicate variables. Technically, we will follow tradition and employ a logical system $\mathcal{HOL}$ based on the simply typed λ-calculus as introduced in [Chu40]; this does not restrict the generality of the methods reported in this article, since the ideas can be carried over. A related logical system is discussed in detail in [And02].

1.2.1 Simply Typed λ-Calculus

To formulate higher order logic, we start with a collection of types $\mathcal{T}$. We assume there are some basic types in $\mathcal{T}$ and that whenever $\alpha, \beta \in \mathcal{T}$, then the function type $(\alpha \to \beta)$ is in $\mathcal{T}$. Furthermore, we assume the types are generated freely, so that $(\alpha_1 \to \beta_1) \equiv (\alpha_2 \to \beta_2)$ implies $\alpha_1 \equiv \alpha_2$ and $\beta_1 \equiv \beta_2$.

$\mathcal{HOL}$-formulae (or *terms*) are built up from a set $\mathcal{V}$ of (typed) variables and a *signature* Σ (a set of typed constants) as *applications* and λ-*abstractions*. We assume the set $\mathcal{V}_\alpha$ of variables of type α is countably infinite for each type α. The set $\mathit{wff}_\alpha(\Sigma)$ of *well-formed formulae* consists of those that can be given a type α so that in all applications, the types of the arguments are the argument types of the function. The type of formula $\mathbf{A}_\alpha$ will be annotated as an index, if it is not clear from the context. We will denote variables with upper-case letters $(X_\alpha, Y, Z, X^1_\beta, X^2_\gamma \ldots)$, constants with lower-case letters $(c_\alpha, f_{\alpha\to\beta}, \ldots)$ and well-formed formulae with upper-case bold letters $(\mathbf{A}_\alpha, \mathbf{B}, \mathbf{C}^1, \ldots)$. Finally, we abbreviate multiple applications and abstractions in a kind of vector notation, so that $\mathbf{A}\overline{\mathbf{U}^k}$ denotes k-fold application (associating to the left) and $\lambda\overline{X^k}.\mathbf{A}$ denotes k-fold λ-abstraction (associating to the right) and we use the square dot '.' as an abbreviation for a pair of brackets, where '.' stands for the left one with its partner as far to the right as is consistent with the bracketing already present in the formula. We may avoid full bracketing of formulas in the remainder if the bracketing structure is clear from the context.

We will use the terms like *free* and *bound* variables or *closed* formulae in their standard meaning and use $free(\mathbf{A})$ for the set of free variables of a formula $\mathbf{A}$. In particular, alphabetic change of names of bound variables is built into $\mathcal{HOL}$: we consider alphabetic variants to be identical (viewing the actual representation as a representative of an alphabetic equivalence class) and use a notion of substitution that avoids variable capture by systematically renaming bound variables. We could also have used de Bruijn's indices [dB72] as a concrete implementation of this approach at the syntax level.

We denote a substitution that instantiates a free variable X with a formula $\mathbf{A}$ with $[\mathbf{A}/X]$ and write $\sigma, [\mathbf{A}/X]$ for the substitution that is identical with σ but instantiates X with $\mathbf{A}$. For any term $\mathbf{A}$ we denote by $\mathbf{A}[\mathbf{B}]_p$ the term resulting by replacing the subterm at position p in $\mathbf{A}$ by $\mathbf{B}$.

A structural equality relation of $\mathcal{HOL}$ terms is induced by *$\beta\eta$-reduction*

$$(\lambda X.\mathbf{A})\mathbf{B} \to_\beta [\mathbf{B}/X]\mathbf{A} \qquad\qquad (\lambda X.\mathbf{C}X) \to_\eta \mathbf{C}$$

where X is not free in $\mathbf{C}$. It is well-known that the reduction relations β, η, and $\beta\eta$ are terminating and confluent on $\mathit{wff}(\Sigma)$, so that there are unique normal forms (cf. [Bar84] for an introduction). We will denote the β-normal form of a term $\mathbf{A}$ by $\mathbf{A}{\downarrow_\beta}$, and the $\beta\eta$-normal form of $\mathbf{A}$ by $\mathbf{A}{\downarrow_{\beta\eta}}$. If we allow both reduction and expansion steps, we obtain notions of *β-conversion*, *η-conversion*, and *$\beta\eta$-conversion*. We say $\mathbf{A}$ and $\mathbf{B}$ are *β-equal* [*η-equal*, *$\beta\eta$-equal*] (written $\mathbf{A}{\equiv_\beta}\mathbf{B}$ [$\mathbf{A}{\equiv_\eta}\mathbf{B}$, $\mathbf{A}{\equiv_{\beta\eta}}\mathbf{B}$]) when $\mathbf{A}$ is β-convertible [η-convertible, $\beta\eta$-convertible] to $\mathbf{B}$.

1.2.2 Higher Order Logic ($\mathcal{HOL}$)

In $\mathcal{HOL}$, the set of base types is $\{o, \iota\}$ for truth values and individuals. We will call a formula of type o a *proposition*, and a *sentence* if it is closed. We will assume that the signature Σ contains logical constants for *negation* ($\neg_{o\to o}$), *disjunction* ($\vee_{o\to o\to o}$), and *universal quantification* ($\Pi^\alpha_{(\alpha\to o)\to o}$) for each type α. Optionally, Σ may contain *primitive equality* ($=^\alpha_{\alpha\to\alpha\to o}$) for each type α. All other constants are called *parameters*, since the argumentation in this article is parametric in their choice. We only assume that there are closed formulae for both base types, and as a consequence that the semantic domains associated with the types are non-empty.

We write disjunctions and equations, i.e. terms of the form $((\vee\mathbf{A})\mathbf{B})$ or $((=\mathbf{A})\mathbf{B})$, in infix notation as $\mathbf{A} \vee \mathbf{B}$ and $\mathbf{A} = \mathbf{B}$. As we only assume the logical constants $\neg$, $\vee$, and Π^α (and possibly $=^\alpha$) as primitive, we will use formulae of the form $\mathbf{A} \wedge \mathbf{B}$, $\mathbf{A} \Rightarrow \mathbf{B}$, and $\mathbf{A} \Leftrightarrow \mathbf{B}$ as shorthand for the formulae $\neg((\neg\mathbf{A}) \vee (\neg\mathbf{B}))$, and $(\neg\mathbf{A}) \vee \mathbf{B}$, and $(\mathbf{A} \Rightarrow \mathbf{B}) \wedge (\mathbf{B} \Rightarrow \mathbf{A})$, respectively. For each $\mathbf{A} \in \mathit{wff}_o(\Sigma)$, the standard notations $\forall X_\alpha.\mathbf{A}$ and $\exists X_\alpha.\mathbf{A}$ for quantification are regarded as shorthand for $\Pi^\alpha(\lambda X_\alpha.\mathbf{A})$ and $\neg(\Pi^\alpha(\lambda X_\alpha.\neg\mathbf{A}))$. Finally, we extend the vector notation for λ-binders to k-fold quantification: we will use $\forall\overline{X^k}.\mathbf{A}$ and $\exists\overline{X^k}.\mathbf{A}$ in the obvious way.

We often need to distinguish between atomic and non-atomic formulae in $\mathit{wff}_o(\Sigma)$. A non-atomic formula is any formula whose β-normal form is either of the form $\neg\mathbf{A}$, $\mathbf{A} \vee \mathbf{B}$, or $\Pi^\alpha\mathbf{C}$ (where $\mathbf{A}, \mathbf{B} \in \mathit{wff}_o(\Sigma)$ and $\mathbf{C} \in \mathit{wff}_{\alpha\to o}(\Sigma)$). An atomic formula is any other formula in $\mathit{wff}_o(\Sigma)$ – including primitive equations $\mathbf{A} =^\alpha \mathbf{B}$ in case of the presence of primitive equality.

It is matter of folklore that equality can directly be expressed in $\mathcal{HOL}$. A prominent example is the *Leibniz formula* for equality

$$\mathbf{Q}^\alpha := (\lambda X_\alpha Y_\alpha.\forall P_{\alpha\to o}.PX \Rightarrow PY)$$

With this definition, the formula $(\mathbf{Q}^\alpha\mathbf{A}\mathbf{B})$ (expressing equality of two formulae $\mathbf{A}$ and $\mathbf{B}$ of type α) β-reduces to $\forall P_{\alpha\to o}.(P\mathbf{A}) \Rightarrow (P\mathbf{B})$, which can be read as: formulae $\mathbf{A}$ and $\mathbf{B}$ are not equal iff there exists a discerning property P.[1] In other words, $\mathbf{A}$ and $\mathbf{B}$ are equal, if they are indiscernible. We will use the notation $(\mathbf{A} \doteq^\alpha \mathbf{B})$ as shorthand for the β-reduct $\forall P_{\alpha\to o}.(P\mathbf{A}) \Rightarrow (P\mathbf{B})$ of $(\mathbf{Q}^\alpha\mathbf{A}\mathbf{B})$ (where $P \notin \mathit{free}(\mathbf{A}) \cup \mathit{free}(\mathbf{B})$). (Note that $\mathbf{A} \doteq^\alpha \mathbf{B}$ is β-normal iff $\mathbf{A}$ and $\mathbf{B}$ are β-normal. The same holds for $\beta\eta$-conversion.) There are alternative ways to define equality in terms of the logical

[1] Note that this is symmetric by considering complements and hence it is sufficient to use $\Rightarrow$ instead of $\Leftrightarrow$.

connectives ([And02, p. 203]) and the techniques for equality introduced in this article carry over to them (cf. Remark 1.4.4).

In this article we use several different notions of equality. In order to prevent misunderstandings we explain these different notions together with their syntactical representation here:

If we *define* a concept we use $:=$ (e.g. let $\mathcal{D} := \{\mathtt{T}, \mathtt{F}\}$). $\equiv$ represents identity. We refer to a representative of the equality relation on $\mathcal{D}_\alpha$ as an *object* of the *semantical domain* $\mathcal{D}_{\alpha\to\alpha\to o}$ with q^α. Note that we possibly have one, several, or no q^α in $\mathcal{D}_{\alpha\to\alpha\to o}$ for each domain $\mathcal{D}_\alpha$. The remaining two notions are related to syntax. $=^\alpha$ may occur as a *constant symbol* of type $\alpha \to \alpha \to o$ in a signature Σ. Finally, $\doteq^\alpha$ and $\mathbf{Q}^\alpha$ are used for *Leibniz equality* as described above.

1.2.3 Notions of Models for $\mathcal{HOL}$

A model of $\mathcal{HOL}$ is a collection of non-empty domains $\mathcal{D}_\alpha$ for all types α together with a way of interpreting formulae. The model classes discussed in this article will vary in the domains and specifics of the evaluation of formulae. The relationships between these classes of models are depicted as a cube in Figure 1.1. We will discuss the model classes from bottom to top, from the most specific notion of standard models ($\mathfrak{ST}$) to the most general notion of υ-complexes, motivating the respective generalizations as we go along. In Section 1.3, where we develop the theory formally based on the intuitions discussed here, we will proceed the other way around, specializing the notion of a Σ-model more and more.

The symbols in the boxes in Figure 1.1 denote model classes, the symbols labeling the arrows indicate the properties inducing the corresponding specialization, and the ∇-symbols next to the boxes indicate the clauses in the definition of abstract consistency classes (cf. Definition 1.6.5) that are needed to establish a model existence theorem for this particular class of models (cf. Theorem 1.6.35).

Standard- and Henkin Models $[\mathfrak{ST},\mathfrak{H},\mathfrak{M}_{\beta\mathfrak{fb}}]$ A *standard model* ($\mathfrak{ST}$, cf. Definition 1.3.51) for $\mathcal{HOL}$ provides a fixed set $\mathcal{D}_\iota$ of individuals and a set $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$ of truth values. All the domains for the function types are defined inductively: $\mathcal{D}_{\alpha\to\beta}$ is the set of functions $f{:}\mathcal{D}_\alpha \to \mathcal{D}_\beta$. The evaluation function $\mathcal{E}_\varphi$ with respect to an assignment φ of variables is obtained by the standard homomorphic construction that evaluates a λ-abstraction with a function.

One can reconstruct the key idea behind *Henkin model* ($\mathfrak{H}$ isomorphic to $\mathfrak{M}_{\beta\mathfrak{fb}}$, cf. Definitions 1.3.50, and Theorem 1.3.69) by the following observation. If the set $\mathcal{D}_\iota$ is infinite, the set $\mathcal{D}_{\iota\to o}$ of sets of individuals must be uncountably infinite. On the other hand, any semantics of a language with a countable signature that admits sound and complete calculi must have countable models, because of the compactness theorem that comes with a complete calculus. Leon Henkin generalized the class of admissible domains for functional types [Hen50]. 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 (in other words, the domains of function types are rich enough to satisfy comprehension). 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). The generalization to Henkin models restricts the set of valid formulae sufficiently so that all of them can be proven by a Hilbert-style calculus [Hen50].

Of course our picture in Figure 1.1 is not complete here; we can axiomatically require the existence of particular (classes of) functions, e.g. by assuming the description or choice operators. We will not pursue this here; for a detailed discussion of the semantic issues raised by the presence of these logical constants see [And72b]. Note that even though we can consider model classes with

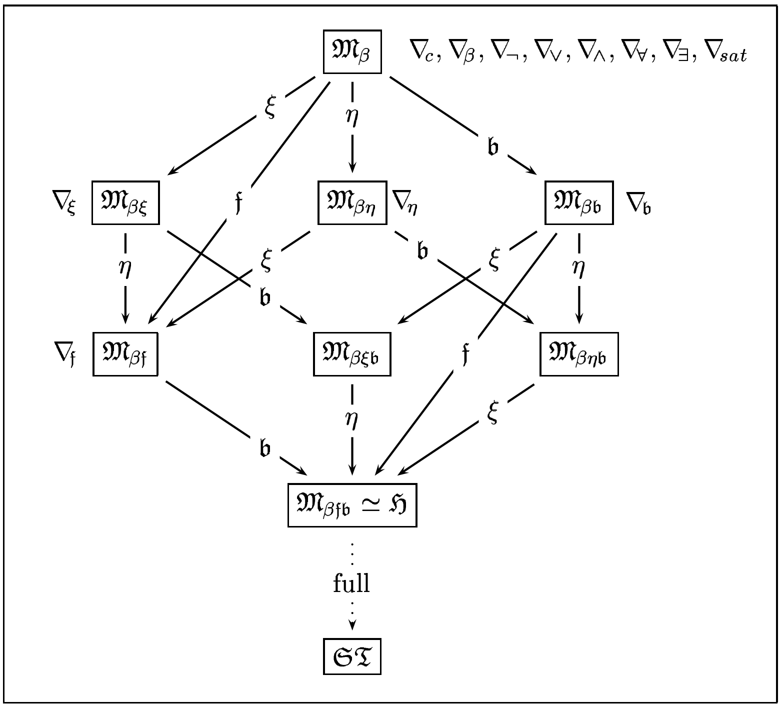

Figure 1.1: The landscape of Higher Order Semantics

richer and richer function spaces, we can never reach standard models where function spaces are full while maintaining complete (recursively axiomatizable) calculi.

Models without Boolean Extensionality [$\mathfrak{M}_\beta$, $\mathfrak{M}_{\beta\xi}$, $\mathfrak{M}_{\beta\eta}$, $\mathfrak{M}_{\beta\mathfrak{f}}$] The next generalization of model classes comes from the fact that we want to have logics where the axiom of Boolean extensionality can fail. For instance, in the semantics of natural language we have so-called verbs and adjectives of "propositional attitude" like *believe* or *obvious*. We may not want to commit ourselves to a logic where the sentence "John believes that Phil is a woodchuck" automatically entails "John believes that Phil is a groundhog" since John might not be aware that "woodchuck" is just another word for "groundhog". The axiom of Boolean extensionality does just that; it states that whenever two propositions are equivalent, they must be equal, and can be substituted for each other. Similarly, the formulae $obvious(\mathbf{O})$ and $obvious(\mathbf{F})$ where $\mathbf{O} := 2 + 2 = 4$ and $\mathbf{F} := \forall x > 2.x^n + y^n = z^n \Rightarrow x = y = z = 0$ should not be equivalent, even if their arguments are. (Both $\mathbf{O}$ and $\mathbf{F}$ are true over the natural numbers, but Fermat's last theorem $\mathbf{F}$ is non-obvious to most people). These phenomena have been studied under the heading of "hyperintensional semantics" in theoretical semantics; see [LP00] for a survey.

To account for this behavior, we have to generalize the class of Henkin models further so that there are counter-models to the examples above. Obviously, this involves weakening the assumption that $\mathcal{D}_o \equiv \{\mathtt{T}, \mathtt{F}\}$ since this entails that the values of $\mathbf{O}$ and $\mathbf{F}$ are identical. We call the assumption that $\mathcal{D}_o$ has two elements property $\mathfrak{b}$. In our Σ-*models* without property $\mathfrak{b}$ ($\mathfrak{M}_\beta, \mathfrak{M}_{\beta\xi}, \mathfrak{M}_{\beta\eta}, \mathfrak{M}_{\beta\mathfrak{f}}$, cf. Definitions 1.3.41 and 1.3.49) we only insist that there is a valuation υ of $\mathcal{D}_o$, i.e., a function $\upsilon\colon \mathcal{D}_o \to \{\mathtt{T}, \mathtt{F}\}$ that is coordinated with the interpretations of the logical constants $\neg$, $\vee$, and Π^α (for each type α). Thus we have a notion of validity: we call a sentence $\mathbf{A}$ valid in such a model if $\upsilon(\mathsf{a}) \equiv \mathtt{T}$, where $\mathsf{a} \in \mathcal{D}_o$ is the value of the sentence $\mathbf{A}$. For example, there is a Σ-model (see Examples 1.5.4 and 1.5.5) where $woodchuck(phil)$, $groundhog(phil)$ and $believe(john, woodchuck(phil))$ are all valid, but $believe(john, groundhog(phil))$ is not. In this model, the value of $woodchuck(phil)$ is different from the value of $groundhog(phil)$ in $\mathcal{D}_o$.

Models without Functional Extensionality [$\mathfrak{M}_\beta$, $\mathfrak{M}_{\beta\eta}$, $\mathfrak{M}_{\beta\xi}$, $\mathfrak{M}_{\beta\mathfrak{b}}$, $\mathfrak{M}_{\beta\eta\mathfrak{b}}$, $\mathfrak{M}_{\beta\xi\mathfrak{b}}$] In mathematics (and as a consequence in most higher order model theories), we assume functional extensionality, which states that two functions are equal, if they return identical values on all arguments. In many applications we want to use a logic that allows a finer-grained modeling of properties of functions. For instance, if we want to model programs as (higher order) functions, we might be interested in intensional[2] properties like run-time complexity. Consider for instance the two functions $\mathbf{I} := \lambda X.X$ and $\mathbf{L} := \lambda X.rev(rev(X))$, where rev is the self-inverse function that reverses the order of elements in a list. While the identity function has constant complexity, the function rev is linear in the length of its argument. As a consequence, even though $\mathbf{L}$ behaves like $\mathbf{I}$ on all inputs, they have different time complexity. A logic with a functionally extensional model theory (which is encoded as property $\mathfrak{f}$, cf. Definitions 1.3.5, 1.3.41 and 1.3.46) would conflate $\mathbf{I}$ and $\mathbf{L}$ semantically and thus hide this difference rendering the logic unsuitable for complexity analysis.

To arrive at a model theory which does not require functional extensionality (which we will call non-functional model theory in the remainder) we need to generalize the notion of domains at function types and evaluation functions. This is because the usual construction already uses sets of (extensional) functions for the domains of function type and the property of functionality to construct values for λ-terms.

We build on the notion of applicative structures (cf. Definition 1.3.1) to define Σ-evaluations (cf. Definition 1.3.17), where the evaluation function is assumed to respect application and β-conversion. In such models, a function is not uniquely determined by its behavior on all possible arguments. Such models can be constructed, for example, by labeling for functions (e.g., a green and a red version of a function f) in order to differentiate between them, even though they are functionally equivalent (cf. Example 1.5.6). Property $\mathfrak{b}$ may or may not hold for non-functional Σ-Models.

We can factor functional extensionality (property $\mathfrak{f}$) into two independent properties, property η and property ξ. A model satisfies property η if it respects η-conversion. A model satisfies property ξ if we can conclude the values of $\lambda X.\mathbf{M}$ and $\lambda X.\mathbf{N}$ are identical whenever the values of $\mathbf{M}$ and $\mathbf{N}$ are identical for any assignment of the variable X. We will show that a model satisfies property $\mathfrak{f}$ iff it satisfies both property η and property ξ (cf. Lemma 1.3.22).

Andrews' Models and υ-complexes [$\mathfrak{M}_\beta$, $\mathfrak{M}_{\beta\eta}$] Peter Andrews has pioneered the construction of non-functional models with his υ-complexes in [And71] based on Kurt Schütte's semi-valuation method [Sch60]. These constructions, where both functional and Boolean extensionality fail, are Σ-models as defined in Definition 1.3.41. (Typically they will not even satisfy the property that Leibniz equality corresponds to identity in the model, but they will have a quotient by Theorem 1.3.62 which does satisfy this property.)

[2] Just as in the linguistic application, the word "intensional" is used as a synonym for "non-extensional" even though totally different properties are intended.

1.2.4 Characterizing the Deductive Power of Calculi

These model classes discussed in the previous section characterize the deductive power of many higher order theorem provers on a semantic level. For example, TPS [ABI+96] can be used in modes in which the deductive power is characterized by $\mathfrak{M}_{\beta\eta}$ (or even $\mathfrak{M}_{\beta}$ if η-conversion is disallowed). Note that in particular TPS is not complete with respect to Henkin models. It is not even complete for $\mathfrak{M}_{\beta\eta\mathfrak{b}}$, although it can be used in modes with some 'extensionality treatment' build into the proof procedure.

The incompleteness of TPS for Henkin models[3] can be seen from the fact that it fails to refute formulae such as $c\mathbf{A}_o \wedge \neg c(\neg\neg\mathbf{A})$, where c is a constant of type $o \to o$, or to prove formulae like $c(\lambda X_\alpha.\mathbf{B}X \wedge \mathbf{A}X) \Rightarrow c(\lambda X_\alpha.\mathbf{A}X \wedge \mathbf{B}X)$, where c is a constant of type $(\alpha \to o) \to o$. The problem in the former example is that the higher order unification algorithm employed by TPS cannot determine that $\mathbf{A}$ and $\neg\neg\mathbf{A}$ denote identical semantic objects (by Boolean extensionality as already mentioned before), and thus returns failure instead of success. In the second example both functional and Boolean extensionality is needed in order to prove the theorem.

[DHK01] discusses a presentation of higher order logic in a first order logic based on an approach called *theorem proving modulo*. It is easy to check that this approach is also incomplete for model classes with property $\mathfrak{b}$. For instance the approach cannot prove the formula

$$\forall P_{\alpha\to o} X_o Y_o.(PX \wedge PY) \Rightarrow P(X \wedge Y)$$

which is valid in Henkin models and which requires $\mathfrak{b}$. As a result, the *theorem proving modulo* approach of representing higher order logic in a first order logic [DHK01] can only be used for logics without Boolean extensionality in its current form.

Model Existence Theorems For all the notions of model classes (except, of course, for standard models, where such a theorem cannot hold for recursively axiomatizable logical systems) we present model existence theorems tying the differentiating conditions of the models to suitable conditions in the abstract consistency classes (cf. Section 1.6.3).

A model existence theorem for a logical system $\mathcal{S}$ (i.e., a logical language $\mathcal{L}_\mathcal{S}$ together with a consequence relation $\models_\mathcal{S} \subseteq \mathcal{L}_\mathcal{S} \times \mathcal{L}_\mathcal{S}$) is a theorem of the form:

> *If a set of sentences Φ of $\mathcal{S}$ is a member of an abstract consistency class Γ, then there exists a $\mathcal{S}$-model for Φ.*

For the proof we can use the classical construction in all cases: abstract consistent sets are extended to Hintikka sets (cf. Section 2.4.1), which induce a valuation on a term structure (cf. Definition 1.3.35). We then take a quotient by the congruence induced by Leibniz equality in the term model.

Completeness of Calculi Given a model existence theorem as described above we can show the completeness of a particular calculus $\mathcal{C}$ (i.e., the derivability relation $\vdash_\mathcal{S} \subseteq \mathcal{L}_\mathcal{S} \times \mathcal{L}_\mathcal{S}$) by proving that the class Γ of sets of sentences Φ that are $\mathcal{C}$-consistent (i.e., cannot be refuted in $\mathcal{C}$) is an abstract consistency class. Then the model existence theorem tells us that $\mathcal{C}$-consistent sets of sentences are satisfiable in $\mathcal{S}$. Now we assume that a sentence $\mathbf{A}$ is valid in $\mathcal{S}$, so $\neg\mathbf{A}$ does not have a $\mathcal{S}$-model and is therefore $\mathcal{C}$-inconsistent. Hence, $\neg\mathbf{A}$ is refutable in $\mathcal{C}$. This shows refutation completeness of $\mathcal{C}$. For many calculi $\mathcal{C}$, this also shows $\mathbf{A}$ is provable, thus establishing completeness of $\mathcal{C}$.

[3]In case the extensionality axioms are not available in the search space. Note that one can add extensionality axioms to the calculus in order to achieve – at least in theory – Henkin completeness. But this increases the search space drastically and is not feasible in practice.

Note that with this argumentation the completeness proof for $\mathcal{C}$ condenses to verifying that Γ is an abstract consistency class, a task that does not refer to $\mathcal{S}$-models. Thus the usefulness of model existence theorems derives from the fact that it replaces the model-theoretic analysis in completeness proofs with the verification of some proof-theoretic conditions. In this respect a model existence theorem is similar to a Herbrand Theorem, but it is easier to generalize to other logic systems like higher order logic. The technique was developed for first order logic by Jaakko Hintikka and Raymond Smullyan [Hin55, Smu63, Smu68].

1.3 Semantics for Higher Order Logic

In this section we will introduce the semantical constructions and discuss their relationships. We will start out by defining applicative structures and Σ-evaluations to give an algebraic semantics for the simply typed λ-calculus. To obtain a model for higher order logic, we use a Σ-valuation to determine whether propositions are true or false.

1.3.1 Applicative Structures

Definition 1.3.1 ((Typed) Applicative Structure)
A collection $\mathcal{D} := \mathcal{D}_{\mathcal{T}} := \{\mathcal{D}_\alpha \mid \alpha \in \mathcal{T}\}$ of non-empty sets $\mathcal{D}_\alpha$, indexed by the set $\mathcal{T}$ of types, is called a *typed collection* (of sets). Let $\mathcal{D}_{\mathcal{T}}$ and $\mathcal{E}_{\mathcal{T}}$ be typed collections, then a collection $\mathsf{f} := \{\mathsf{f}^\alpha : \mathcal{D}_\alpha \to \mathcal{E}_\alpha \mid \alpha \in \mathcal{T}\}$ of functions is called a *typed function* $\mathsf{f} : \mathcal{D}_{\mathcal{T}} \longrightarrow \mathcal{E}_{\mathcal{T}}$. We will write $\mathcal{F}(A;B)$ for the set of functions from A to B and $\mathcal{F}_{\mathcal{T}}(\mathcal{D}_{\mathcal{T}};\mathcal{E}_{\mathcal{T}})$ for the set of typed functions. In the following we will also use the notion of a typed function extended to the n-ary case in the obvious way.

We call the pair $(\mathcal{D}, @)$ a (typed) *applicative structure* if $\mathcal{D} \equiv \mathcal{D}_{\mathcal{T}}$ is a typed collection of sets and

$$@ := \{@^{\alpha\beta} : \mathcal{D}_{\alpha\to\beta} \times \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta \mid \alpha, \beta \in \mathcal{T}\}$$

Each (non-empty) set $\mathcal{D}_\alpha$ is called the *domain* of type α and the family of functions @ is called the *application operator*. We write simply $\mathsf{f}@\mathsf{a}$ for $\mathsf{f}@^{\alpha\beta}\mathsf{a}$ when $\mathsf{f} \in \mathcal{D}_{\alpha\to\beta}$ and $\mathsf{a} \in \mathcal{D}_\alpha$ are clear in context.

Remark 1.3.2 Often an applicative structure is defined to also include an interpretation of the constants in a given signature (for example, in [Mit96]). We prefer this signature-independent definition (as in [HL99]) for our purposes.

Remark 1.3.3 (Currying) The application operator @ in an applicative structure is an abstract version of function application. It is no restriction to exclusively use a binary application operator, which corresponds to unary function application, since we can define higher-arity application operators from the binary one by setting $\mathsf{f}@(\mathsf{a}^1, \ldots, \mathsf{a}^n) := (\ldots(\mathsf{f}@\mathsf{a}^1)\ldots@\mathsf{a}^n)$ ("Currying").

Definition 1.3.4 (Frame) An applicative structure $(\mathcal{D}, @)$ is called a *frame*, if $\mathcal{D}_{\alpha\to\beta} \subseteq \mathcal{F}(\mathcal{D}_\alpha;\mathcal{D}_\beta)$ and $@^{\alpha\beta}$ is application for functions for all types α and β.

Definition 1.3.5 (Functional/Full/Standard Applicative Structures)
Let $\mathcal{A} := (\mathcal{D}, @)$ be an applicative structure. We say $\mathcal{A}$ is *functional* if for all types α and β and objects $\mathsf{f}, \mathsf{g} \in \mathcal{D}_{\alpha\to\beta}$, we have $\mathsf{f} \equiv \mathsf{g}$ whenever $\mathsf{f}@\mathsf{a} \equiv \mathsf{g}@\mathsf{a}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$[4] We say $\mathcal{A}$ is *full* if

[4] This is called "extensional" in [Mit96]. We use the term "functional" to distinguish it from other forms of extensionality.

for all types α and β and every function $f: \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$ there is an object $\mathsf{f} \in \mathcal{D}_{\alpha\to\beta}$ such that $\mathsf{f}@\mathsf{a} \equiv f(\mathsf{a})$ for every $\mathsf{a} \in \mathcal{D}_\alpha$. Finally, we say $\mathcal{A}$ is *standard* if it is a frame and $\mathcal{D}_{\alpha\to\beta} \equiv \mathcal{F}(\mathcal{D}_\alpha; \mathcal{D}_\beta)$ for all types α and β. Note that these definitions impose restrictions on the domains for function types only.

Remark 1.3.6 It is easy to show that every frame is functional. Furthermore, an applicative structure is standard iff it is a full frame.

Example 1.3.7 (Applicative Singleton Structure) We choose a single element a and define $\mathcal{D}_\alpha := \{\mathsf{a}\}$ for all types α. The pair $(\mathcal{D}_\mathcal{T}, @^\mathsf{a})$, where $\mathsf{a}@^\mathsf{a}\mathsf{a} = \mathsf{a}$ is a (trivial) example of a functional applicative structure. It is called the *singleton applicative structure*.

Example 1.3.8 (Applicative Term Structures) If we define $\mathbf{A}@\mathbf{B} := (\mathbf{AB})$ for $\mathbf{A} \in \mathit{wff}_{\alpha\to\beta}(\Sigma)$ and $\mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$, then $@: \mathit{wff}_{\alpha\to\beta}(\Sigma) \times \mathit{wff}_\alpha(\Sigma) \longrightarrow \mathit{wff}_\beta(\Sigma)$ is a total function. Thus $(\mathit{wff}(\Sigma), @)$ is an applicative structure. The intuition behind this example is that we can think of the formula $\mathbf{A} \in \mathit{wff}_{\alpha\to\beta}(\Sigma)$ as a function $\mathbf{A}: \mathit{wff}_\alpha(\Sigma) \longrightarrow \mathit{wff}_\beta(\Sigma)$ that maps $\mathbf{B}$ to $(\mathbf{AB})$.

Analogously, we can define the applicative structure $(\mathit{cwff}(\Sigma), @)$ of closed formulae (when we ensure Σ contains enough constants so that $\mathit{cwff}_\alpha(\Sigma)$ is non-empty for all types α).

Definition 1.3.9 (Homomorphism) Let $\mathcal{A}^1 := (\mathcal{D}^1, @^1)$ and $\mathcal{A}^2 := (\mathcal{D}^2, @^2)$ be applicative structures. A *homomorphism* from $\mathcal{A}^1$ to $\mathcal{A}^2$ is a typed function $\kappa: \mathcal{D}^1 \longrightarrow \mathcal{D}^2$ such that for all types $\alpha, \beta \in \mathcal{T}$, all $\mathsf{f} \in \mathcal{D}^1_{\alpha\to\beta}$, and $\mathsf{a} \in \mathcal{D}^1_\alpha$ we have $\kappa(\mathsf{f})@^2\kappa(\mathsf{a}) \equiv \kappa(\mathsf{f}@^1\mathsf{a})$. We write $\kappa: \mathcal{A}^1 \longrightarrow \mathcal{A}!2$. The two applicative structures $\mathcal{A}^1$ and $\mathcal{A}^2$ are called *isomorphic* if there are homomorphisms $i: \mathcal{A}^1 \longrightarrow \mathcal{A}^2$ and $j: \mathcal{A}^2 \longrightarrow \mathcal{A}^1$ which are mutually inverse at each type.

The most important method for constructing structures (and models) with given properties in this article is well-known for algebraic structures and consists of building a suitable congruence and passing to the quotient structure. We will now develop the formal basis for it.

Definition 1.3.10 (Applicative Structure Congruences) Let $\mathcal{A} := (\mathcal{D}, @)$ be an applicative structure. A typed equivalence relation $\sim$ is called a *congruence* on $\mathcal{A}$ iff for all $\mathsf{f}, \mathsf{f}' \in \mathcal{D}_{\alpha\to\beta}$ and $\mathsf{a}, \mathsf{a}' \in \mathcal{D}_\alpha$ (for any types α and β), $\mathsf{f} \sim \mathsf{f}'$ and $\mathsf{a} \sim \mathsf{a}'$ imply $\mathsf{f}@\mathsf{a} \sim \mathsf{f}'@\mathsf{a}'$.

The *equivalence class* $[\![\mathsf{a}]\!]_\sim$ of $\mathsf{a} \in \mathcal{D}_\alpha$ *modulo* $\sim$ is the set of all $\mathsf{a}' \in \mathcal{D}_\alpha$, such that $\mathsf{a} \sim \mathsf{a}'$. A congruence $\sim$ is called *functional* iff for all types α and β and $\mathsf{f}, \mathsf{g} \in \mathcal{D}_{\alpha\to\beta}$, we have $\mathsf{f} \sim \mathsf{g}$ whenever $\mathsf{f}@\mathsf{a} \sim \mathsf{g}@\mathsf{a}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$.

Lemma 1.3.11 *The β-equality and $\beta\eta$-equality relations $\equiv_\beta$ and $\equiv_{\beta\eta}$ are congruences on the applicative structures $\mathit{wff}(\Sigma)$ and $\mathit{cwff}(\Sigma)$.*

Proof: The congruence properties are a direct consequence of the fact that $\beta\eta$-reduction rules are defined to act on subterm positions. $\square$

Definition 1.3.12 (Quotient Applicative Structure) Let $\mathcal{A} := (\mathcal{D}, @)$ be an applicative structure, $\sim$ a congruence on $\mathcal{A}$, and $\mathcal{D}^\sim_\alpha := \{[\![\mathsf{a}]\!]_\sim \mid \mathsf{a} \in \mathcal{D}_\alpha\}$. Furthermore, let $@^\sim$ be defined by $[\![\mathsf{f}]\!]_\sim @^\sim [\![\mathsf{a}]\!]_\sim := [\![\mathsf{f}@\mathsf{a}]\!]_\sim$. (To see that this definition only depends on equivalence classes of $\sim$, consider $\mathsf{f}' \in [\![\mathsf{f}]\!]_\sim$ and $\mathsf{a}' \in [\![\mathsf{a}]\!]_\sim$. Then $\mathsf{f} \sim \mathsf{f}'$ and $\mathsf{a} \sim \mathsf{a}'$ imply $\mathsf{f}@\mathsf{a} \sim \mathsf{f}'@\mathsf{a}'$. Thus, $[\![\mathsf{f}@\mathsf{a}]\!]_\sim \equiv [\![\mathsf{f}'@\mathsf{a}']\!]_\sim$. So, $@^\sim$ is well-defined.) $\mathcal{A}/_\sim := (\mathcal{D}^\sim, @^\sim)$ is also an applicative structure. We call $\mathcal{A}/_\sim$ the *quotient structure* of $\mathcal{A}$ for the relation $\sim$ and the typed function $\pi_\sim: \mathcal{A} \longrightarrow \mathcal{A}/_\sim$ that maps a to $[\![\mathsf{a}]\!]_\sim$ its *canonical projection*.

Theorem 1.3.13 *Let $\mathcal{A}$ be an applicative structure and let $\sim$ be a congruence on $\mathcal{A}$, then the canonical projection $\pi_\sim$ is a surjective homomorphism. Furthermore, $\mathcal{A}/_\sim$ is functional iff $\sim$ is functional.*

Proof: Let $\mathcal{A} := (\mathcal{D}, @)$ be an applicative structure. To convince ourselves that $\pi_\sim$ is indeed a surjective homomorphism, we note that $\pi_\sim$ is surjective by the definition of $\mathcal{D}^\sim$. To see that $\pi_\sim$ is a homomorphism let $\mathsf{f} \in \mathcal{D}_{\alpha\to\beta}$, and $\mathsf{a} \in \mathcal{D}_\beta$, then $\pi_\sim(\mathsf{f})@^\sim\pi_\sim(\mathsf{a}) \equiv [\![\mathsf{f}]\!]_\sim @^\sim [\![\mathsf{a}]\!]_\sim \equiv [\![\mathsf{f}@\mathsf{a}]\!]_\sim \equiv \pi_\sim(\mathsf{f}@\mathsf{a})$.

The quotient construction collapses $\sim$ to identity, so functionality of $\sim$ is equivalent to functionality of $\mathcal{A}/_\sim$. Formally, suppose $[\![\mathsf{f}]\!]_\sim$ and $[\![\mathsf{g}]\!]_\sim$ are elements of $\mathcal{D}^\sim_{\alpha\to\beta}$ such that $[\![\mathsf{f}]\!]_\sim @^\sim [\![\mathsf{a}]\!]_\sim \equiv [\![\mathsf{g}]\!]_\sim @^\sim [\![\mathsf{a}]\!]_\sim$ for every $[\![\mathsf{a}]\!]_\sim$ in $\mathcal{D}^\sim_\alpha$. This is equivalent to $[\![\mathsf{f}@\mathsf{a}]\!]_\sim \equiv [\![\mathsf{g}@\mathsf{s}]\!]_\sim$ for every $\mathsf{a} \in \mathcal{D}_\alpha$ and hence $\mathsf{f}@\mathsf{a} \sim \mathsf{g}@\mathsf{a}$ for all $\mathsf{a} \in \mathcal{D}_\alpha$. By functionality of $\sim$, we have $\mathsf{f} \sim \mathsf{g}$. That is, $[\![\mathsf{f}]\!]_\sim \equiv [\![\mathsf{g}]\!]_\sim$. $\Box$

Lemma 1.3.14 *$\equiv_{\beta\eta}$ is a functional congruence on wff(Σ). If Σ_α is infinite for all types $\alpha \in \mathcal{T}$, then $\equiv_{\beta\eta}$ is also functional on cwff(Σ).*

Proof: It is straightforward to verify that $\equiv_{\beta\eta}$ is a congruence relation. To show functionality let $\mathbf{A}, \mathbf{B} \in \mathit{wff}_{\gamma\to\alpha}(\Sigma)$ such that $\mathbf{AC}\equiv_{\beta\eta}\mathbf{BC}$ for all $\mathbf{C} \in \mathit{wff}_\gamma(\Sigma)$ be given. In particular, for any variable $X \in \mathcal{V}_\gamma$ that is not free in $\mathbf{A}$ or $\mathbf{B}$, we have $\mathbf{A}X\equiv_{\beta\eta}\mathbf{B}X$ and $\lambda X.\mathbf{A}X\equiv_{\beta\eta}\lambda X.\mathbf{B}X$. By definition we have $\mathbf{A}\equiv_\eta\lambda X_\gamma.\mathbf{A}X\equiv_{\beta\eta}\lambda X_\gamma.\mathbf{B}X\equiv_\eta\mathbf{B}$.

To show functionality of $\beta\eta$-equality on closed formulae, suppose $\mathbf{A}$ and $\mathbf{B}$ are closed. With the same variable X as above, let $\mathbf{M}$ and $\mathbf{N}$ be the $\beta\eta$-normal forms of $\mathbf{A}X$ and $\mathbf{B}X$, respectively. We cannot conclude that $\mathbf{M} \equiv \mathbf{N}$ since X is not a closed term. Instead, choose a constant $c_\gamma \in \Sigma_\gamma$ that does not occur in $\mathbf{A}$ or $\mathbf{B}$. (Such a constant must exist, since we have assumed that Σ_γ is infinite.) An easy induction on the length of the $\beta\eta$-reduction sequence from $\mathbf{A}X$ to $\mathbf{M}$ shows that c does not occur in $\mathbf{M}$ and $\mathbf{A}c \equiv [c/X](\mathbf{A}X)$ $\beta\eta$-reduces to $[c/X]\mathbf{M}$. Similarly, c does not occur in $\mathbf{N}$ and $\mathbf{B}c$ $\beta\eta$-reduces to $[c/X]\mathbf{N}$. Since c is a constant, substituting c for X cannot introduce new redexes. So, easy inductions on the sizes of $\mathbf{M}$ and $\mathbf{N}$ show $[c/X]\mathbf{M}$ and $[c/X]\mathbf{N}$ are $\beta\eta$-normal. By assumption, we know $\mathbf{A}c\equiv_{\beta\eta}\mathbf{B}c$. Since normal forms are unique, we must have $[c/X]\mathbf{M} \equiv [c/X]\mathbf{N}$. Using the fact that c does not occur in either $\mathbf{M}$ or $\mathbf{N}$, an easy induction on the size of $\mathbf{M}$ shows $\mathbf{M} \equiv \mathbf{N}$. So, we have $\mathbf{A}\equiv_\eta\lambda X_\gamma.\mathbf{A}X\equiv_{\beta\eta}\lambda X_\gamma.\mathbf{M} \equiv \lambda X_\gamma.\mathbf{N}\equiv_{\beta\eta}\lambda X_\gamma.\mathbf{B}X\equiv_\eta\mathbf{B}$ $\Box$

Remark 1.3.15 Suppose we have a signature Σ with a single constant c_ι. In this case, c is the only closed $\beta\eta$-normal form of type ι. Since $\lambda X.X \not\equiv_{\beta\eta} \lambda X.c$ even though $(\lambda X.X)c\equiv_{\beta\eta}c\equiv_{\beta\eta}(\lambda X.c)c$ we have a counterexample to functionality of $\equiv_{\beta\eta}$ on *cwff*(Σ). The problem here is that we do not have another constant d_ι to distinguish the two functions. In *wff*(Σ) we could always use a variable.

Remark 1.3.16 (Assumptions on Σ) From now on, we assume Σ_α to be infinite for each type α. Furthermore, we assume there is a particular cardinal $\aleph_s$ such that Σ_α has cardinality $\aleph_s$ for every type α. Since $\mathcal{V}$ is countable, this implies $\mathit{wff}_\alpha(\Sigma)$ and $\mathit{cwff}_\alpha(\Sigma)$ have cardinality $\aleph_s$ for each type α. Also, whether or not primitive equality is included in the signature, there can only be finitely many logical constants in Σ_α for each particular type α. Thus, the cardinality of the set of parameters in Σ_α is also $\aleph_s$. In the countable case, of course, $\aleph_s$ is $\aleph_0$.

1.3.2 Σ-Evaluations

Σ-evaluations are applicative structures with a notion of evaluation for well-formed formulae in *wff*(Σ).

Definition 1.3.17 (Σ-Evaluation) Let $\mathcal{A} := (\mathcal{D}, @)$ be an applicative structure. A typed function $\varphi: \mathcal{V} \longrightarrow \mathcal{D}$ is called a *variable assignment* into $\mathcal{A}$. Given a variable assignment φ, variable X_α, and value $\mathsf{a} \in \mathcal{D}_\alpha$, we use $\varphi, [\mathsf{a}/X]$ to denote the variable assignment with $(\varphi, [\mathsf{a}/X])(X) \equiv \mathsf{a}$ and $(\varphi, [\mathsf{a}/X])(Y) \equiv \varphi(Y)$ for variables Y other than X.

Let $\mathcal{E}: \mathcal{F}_{\mathcal{T}}(\mathcal{V}; \mathcal{D}) \times \mathit{wff}(\Sigma) \longrightarrow \mathcal{D}$ be a total binary typed function, where $\mathcal{F}_{\mathcal{T}}(\mathcal{V}; \mathcal{D})$ is the set of variable assignments. We will consider the first argument of $\mathcal{E}$ as a parameter to $\mathcal{E}$ and write $\mathcal{E}_\varphi$ for the typed unary function $\mathcal{E}(\varphi, \cdot): \mathit{wff}(\Sigma) \longrightarrow \mathcal{D}$ that maps terms in $\mathit{wff}(\Sigma)$ to objects in $\mathcal{D}$. $\mathcal{E}$ is called an *evaluation function* for $\mathcal{A}$ if for any assignment φ into $\mathcal{A}$, we have

1. $\mathcal{E}_\varphi\big|_{\mathcal{V}} \equiv \varphi$.
2. $\mathcal{E}_\varphi(\mathbf{FA}) \equiv \mathcal{E}_\varphi(\mathbf{F}) @ \mathcal{E}_\varphi(\mathbf{A})$ 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. $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\psi(\mathbf{A})$ for any type α and $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, whenever φ and ψ coincide on $free(\mathbf{A})$.
4. $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{A}{\downarrow_\beta})$ for all $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$.

We call $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ a Σ-*evaluation* if $(\mathcal{D}, @)$ is an applicative structure and $\mathcal{E}$ is an evaluation function for $(\mathcal{D}, @)$. We call $\mathcal{E}_\varphi(\mathbf{A}_\alpha) \in \mathcal{D}_\alpha$ the *denotation* of $\mathbf{A}_\alpha$ in $\mathcal{J}$ for φ. (Note that since $\mathcal{E}$ is a function, the denotation in $\mathcal{J}$ is unique. However, for a given applicative structure $\mathcal{A}$, there may be many possible evaluation functions.)

If $\mathbf{A}$ is a closed formula, then $\mathcal{E}_\varphi(\mathbf{A})$ is independent of φ, since $free(\mathbf{A}) = \emptyset$. In these cases we sometimes drop the reference to φ from $\mathcal{E}_\varphi(\mathbf{A})$ and simply write $\mathcal{E}(\mathbf{A})$.

We call a Σ-evaluation $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ *functional* [*full*, *standard*] if the applicative structure $(\mathcal{D}, @)$ is *functional* [*full*, *standard*]. We say $\mathcal{J}$ is a Σ-evaluation over a frame if $(\mathcal{D}, @)$ is a frame.

Σ-evaluations generalize Σ-evaluations over frames, which are the basis for Henkin models, to the non-functional case. The existence of an evaluation function that meets the conditions above seems to be the weakest situation where one would like to speak of a model. We cannot in general assume the evaluation function is uniquely determined by its values on constants as this requires functionality.

Remark 1.3.18 (Σ-Evaluations respect β-Equality) Let $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation and $\mathbf{A} \equiv_\beta \mathbf{B}$. For all assignments φ into $(\mathcal{D}, @)$, we have $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{A}{\downarrow_\beta}) \equiv \mathcal{E}_\varphi(\mathbf{B}{\downarrow_\beta}) \equiv \mathcal{E}_\varphi(\mathbf{B})$.

We will consider two weaker notions of functionality. These forms are often discussed in the literature (cf. [HS86]).

Definition 1.3.19 (Weakly Functional Evaluations) Let $\mathcal{J} \equiv (\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation. We say $\mathcal{J}$ is η-*functional* if $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{A}{\downarrow_{\beta\eta}})$ 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, $\mathcal{E}_\varphi(\lambda X_\alpha . \mathbf{M}_\beta) \equiv \mathcal{E}_\varphi(\lambda X_\alpha . \mathbf{N}_\beta)$ whenever $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N})$ for every $\mathsf{a} \in \mathcal{D}_\alpha$.

We will now establish that functionality is equivalent to η-functionality and ξ-functionality combined. We prepare for this by first proving two lemmas about functional Σ-evaluations.

Lemma 1.3.20 *Let $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ be a functional Σ-evaluation. For any assignment φ into $\mathcal{J}$ and $\mathbf{F} \in \mathit{wff}_{\alpha \to \beta}(\Sigma)$ where $X_\alpha \notin free(\mathbf{F})$, we have $\mathcal{E}_\varphi(\lambda X_\alpha . \mathbf{F} X) \equiv \mathcal{E}_\varphi(\mathbf{F})$.*

Proof: Let $\mathsf{a} \in \mathcal{D}_\alpha$ be given. Since $X_\alpha \notin free(\mathbf{F})$, we have $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{F}) \equiv \mathcal{E}_\varphi(\mathbf{F})$. Since $\mathcal{E}$ respects β-equality (cf. Remark 1.3.18), we have $\mathcal{E}_{\varphi,[\mathsf{a}/X]}((\lambda X.\mathbf{F}X)X) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{F}X)$. Thus, we can compute

$$\mathcal{E}_\varphi(\lambda X.\mathbf{F}X)@\mathsf{a} \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}((\lambda X.\mathbf{F}X)X) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{F}X) \equiv \mathcal{E}_\varphi(\mathbf{F})@\mathsf{a}$$

Generalizing over a, we conclude $\mathcal{E}_\varphi(\lambda X.\mathbf{F}X) \equiv \mathcal{E}_\varphi(\mathbf{F})$ by functionality. $\square$

Lemma 1.3.21 *Let $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ be a functional Σ-evaluation. If a formula $\mathbf{A}$ η-reduces to $\mathbf{B}$ in one step, then for any assignment φ into $\mathcal{J}$, $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{B})$.*

Proof: We prove this by induction on the structure of the term $\mathbf{A}$. For the base case when $\mathbf{A}$ is the η-redex which is reduced, we apply Lemma 1.3.20. When $\mathbf{A} \equiv (\mathbf{FC})$, then the η-reduction either occurs in $\mathbf{F}$ or $\mathbf{C}$. So, $\mathbf{B} \equiv (\mathbf{GD})$ where $\mathbf{F}$ η-reduces to $\mathbf{G}$ in one step (or $\mathbf{G} \equiv \mathbf{F}$) and $\mathbf{D} \equiv \mathbf{C}$ (or $\mathbf{C}$ η-reduces to $\mathbf{D}$ in one step). So, by induction we have $\mathcal{E}_\varphi(\mathbf{F}) \equiv \mathcal{E}_\varphi(\mathbf{G})$ and $\mathcal{E}_\varphi(\mathbf{C}) \equiv \mathcal{E}_\varphi(\mathbf{D})$. It follows that $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{B})$.

When $\mathbf{A}$ is a λ-abstraction, we must use functionality. Suppose for some type α, $\mathbf{A} \equiv (\lambda X_\alpha.\mathbf{C})$ (and this is not the η-redex reduced to obtain $\mathbf{B}$). Then $\mathbf{B} \equiv (\lambda X_\alpha \mathbf{D})$ where $\mathbf{C}$ η-reduces in one step to $\mathbf{D}$. By induction hypothesis, for any $\mathsf{a} \in \mathcal{D}_\alpha$, $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{C}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{D})$. Since $\mathcal{E}$ is an evaluation function, we have

$$\begin{aligned}
\mathcal{E}_\varphi(\lambda X.\mathbf{C})@\mathsf{a} &\equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}((\lambda X.\mathbf{C})X) \\
&\equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{C}) \\
&\equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{D}) \\
&\equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}((\lambda X.\mathbf{D})X) \\
&\equiv \mathcal{E}_\varphi(\lambda X.\mathbf{D})@\mathsf{a}
\end{aligned}$$

By functionality, $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\lambda X.\mathbf{C}) \equiv \mathcal{E}_\varphi(\lambda X.\mathbf{D}) \equiv \mathcal{E}_\varphi(\mathbf{B})$. $\square$

Lemma 1.3.22 (Functionality) *Let $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation. Then $\mathcal{J}$ is functional iff it is both η-functional and ξ-functional.*

Proof: The fact that functionality implies η-functionality now follows from a simple induction on the number of $\beta\eta$-reduction steps using Lemma 1.3.21 and Remark 1.3.18.

To show functionality implies ξ-functionality, let $\mathbf{M}, \mathbf{N} \in \mathit{wff}_\beta(\Sigma)$, an assignment φ and a variable X_α be given. Suppose $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N})$ for every $\mathsf{a} \in \mathcal{D}_\alpha$. We need to show $\mathcal{E}_\varphi(\lambda X.\mathbf{M}) \equiv \mathcal{E}_\varphi(\lambda X.\mathbf{N})$. This follows from functionality since

$$\begin{aligned}
\mathcal{E}_\varphi(\lambda X.\mathbf{M})@\mathsf{a} &\equiv \mathcal{E}_{\varphi,[\mathsf{a},X]}((\lambda X.\mathbf{M})X) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \\
&\equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N}) \equiv \mathcal{E}_{\varphi,[\mathsf{a},X]}((\lambda X.\mathbf{N})X) \equiv \mathcal{E}_\varphi(\lambda X.\mathbf{N})@\mathsf{a}
\end{aligned}$$

for every $\mathsf{a} \in \mathcal{D}_\alpha$.

To show functionality from η-functionality and ξ-functionality, let $\mathsf{f}, \mathsf{g} \in \mathcal{D}_{\alpha\to\beta}$ such that $\mathsf{f}@\mathsf{a} \equiv \mathsf{g}@\mathsf{a}$ for all $\mathsf{a} \in \mathcal{D}_\alpha$ be given. We need to show that $\mathsf{f} \equiv \mathsf{g}$. Let $F_{\alpha\to\beta}$, $G_{\alpha\to\beta}$ and X_α be variables and φ be any assignment such that $\varphi(F) \equiv \mathsf{f}$ and $\varphi(G) \equiv \mathsf{g}$. Then for any $\mathsf{a} \in \mathcal{D}_\alpha$ we have $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(FX) \equiv \mathsf{f}@\mathsf{a} \equiv \mathsf{g}@\mathsf{a} \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(GX)$, and thus $\mathcal{E}_\varphi(\lambda X.FX) \equiv \mathcal{E}_\varphi(\lambda X.GX)$ by ξ-functionality. Hence,

$$\mathsf{f} \equiv \mathcal{E}_\varphi(F) \equiv \mathcal{E}_\varphi(\lambda X.FX) \equiv \mathcal{E}_\varphi(\lambda X.GX) \equiv \mathcal{E}_\varphi(G) \equiv \mathsf{g}$$

by η-functionality. $\square$

Lemma 1.3.23 (ξ-Functionality and Replacement) *Let $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ be a ξ-functional Σ-evaluation and $\mathbf{B}, \mathbf{C} \in \mathit{wff}_\beta(\Sigma)$. Suppose $\mathcal{E}_\varphi(\mathbf{B}) \equiv \mathcal{E}_\varphi(\mathbf{C})$ for every assignment φ into $\mathcal{J}$. Then for all formulae $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, positions p, and assignments φ into $\mathcal{J}$, $\mathcal{E}_\varphi(\mathbf{A}[\mathbf{B}]_p) \equiv \mathcal{E}_\varphi(\mathbf{A}[\mathbf{C}]_p)$*

Proof: We show the assertion by an induction on the structure of $\mathbf{A}$. If $\mathbf{A}$ is a constant or a variable, then p must be the empty (i.e., top) position and we have

$$\mathcal{E}_\varphi(\mathbf{A}[\mathbf{B}]_p) \equiv \mathcal{E}_\varphi(\mathbf{B}) \equiv \mathcal{E}_\varphi(\mathbf{C}) \equiv \mathcal{E}_\varphi(\mathbf{A}[\mathbf{C}]_p)$$

If $\mathbf{A}$ is an application $\mathbf{FD}$, we have to consider two cases: $\mathbf{A}[\mathbf{B}]_p = \mathbf{F}[\mathbf{B}]_q\mathbf{D}$ and $\mathbf{A}[\mathbf{B}]_p = \mathbf{F}(\mathbf{D}[\mathbf{B}]_r)$ for some positions q and r. Since the second case is analogous we only show the first case. By inductive hypothesis we have

$$\begin{aligned}
\mathcal{E}_\varphi(\mathbf{A}[\mathbf{B}]_p) &\equiv \mathcal{E}_\varphi(\mathbf{F}[\mathbf{B}]_q\mathbf{D}) \\
&\equiv \mathcal{E}_\varphi(\mathbf{F}[\mathbf{B}]_q)@\mathcal{E}_\varphi(\mathbf{D}) \\
&\equiv \mathcal{E}_\varphi(\mathbf{F}[\mathbf{C}]_q)@\mathcal{E}_\varphi(\mathbf{D}) \\
&\equiv \mathcal{E}_\varphi(\mathbf{F}[\mathbf{C}]_q\mathbf{D}) \\
&\equiv \mathcal{E}_\varphi(\mathbf{A}[\mathbf{C}]_p)
\end{aligned}$$

If $\mathbf{A}[\mathbf{B}]_p = \lambda X_\gamma.\mathbf{D}[\mathbf{B}]_q$, then we get the assertion from ξ-functionality. By the inductive hypothesis, we know $\mathcal{E}_\psi(\mathbf{D}[\mathbf{B}]_q) \equiv \mathcal{E}_\psi(\mathbf{D}[\mathbf{C}]_p)$ for every assignment ψ. In particular, for any assignment φ and $\mathsf{c} \in \mathcal{D}_\gamma$, we have $\mathcal{E}_{\varphi,[\mathsf{c}/X]}(\mathbf{D}[\mathbf{B}]_q) \equiv \mathcal{E}_{\varphi,[\mathsf{c}/X]}(\mathbf{D}[\mathbf{C}]_p)$. By ξ-functionality, we have

$$\mathcal{E}_\varphi(\mathbf{A}[\mathbf{B}]_p) \equiv \mathcal{E}_\varphi(\lambda X.\mathbf{D}[\mathbf{B}]_q) \equiv \mathcal{E}_\varphi(\lambda X.\mathbf{D}[\mathbf{C}]_q) \equiv \mathcal{E}_\varphi(\mathbf{A}[\mathbf{C}]_p)$$

Thus we have completed all the cases and proven the assertion. $\Box$

A key property of Σ-evaluations we will need is a *Substitution-Value Lemma*. This states that $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$ for any assignment φ, types α and β, formulae $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$ and $\mathbf{B} \in \mathit{wff}_\beta(\Sigma)$, and variables X_β. A natural way to attempt the proof of this property is by induction on $\mathbf{A}$. However, such a proof fails at the λ-abstraction step. This is a general problem with trying to prove properties of evaluations since many objects in $\mathcal{D}_{\alpha\to\beta}$ may represent the same function from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$. Fortunately, there is a way to use combinators to reduce such inductions to terms which only have very special λ-abstractions.

Definition 1.3.24 (SK-Combinatory Formulae) For all types α, β, and γ, we define two families of closed formulae we call *combinators*:

$\mathbf{K}_{\alpha\to\beta\to\alpha} := \lambda X_\alpha Y_\beta.X$

$\mathbf{S}_{(\alpha\to\beta\to\gamma)\to(\alpha\to\beta)\to\alpha\to\gamma} := \lambda U_{\alpha\to\beta\to\gamma} V_{\alpha\to\beta} W_\alpha.(UW(VW))$

We define the set of $\mathbf{SK}$-*combinatory formulae* to be the least subset of $\bigcup_{\alpha\in\mathcal{T}} \mathit{wff}_\alpha(\Sigma)$ containing every $\mathbf{K}$ and $\mathbf{S}$, every constant $c \in \Sigma$, every variable, that is closed under application.

As shown in [And72b], every formula can be β-expanded to an $\mathbf{SK}$-combinatory formula.

Lemma 1.3.25 *For every type α and $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, there is an $\mathbf{SK}$-combinatory formula $\mathbf{A}' \in \mathit{wff}_\alpha(\Sigma)$ such that $\mathbf{A}'$ β-reduces to $\mathbf{A}$.*

Proof: See Proposition 1 in [And72b]. The main difference to this setup is the signature, and this plays no role in the proof. $\Box$

Lemma 1.3.26 (Substitution-Value for SK Combinatory Formulae)
Let $(\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation, β be a type, X_β be a variable, and $\mathbf{B} \in wff_\beta(\Sigma)$ be a formula. For every **SK**-*combinatory formula $\mathbf{A} \in wff_\alpha(\Sigma)$, we have $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$.*

Proof: The proof is by a straightforward induction on the definition of the set of **SK**-combinatory formulae. If $\mathbf{A}$ is the variable X_β, then $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(X) \equiv \mathcal{E}_\varphi(\mathbf{B}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]X)$. If $X \notin free(\mathbf{A})$ (e.g., $\mathbf{A}$ is a variable other than X_β, a constant in Σ, or a combinator), then $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$. Finally, if $\mathbf{A}$ is an application of the form $\mathbf{FC}$, then

$$\begin{aligned}
\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{FC}) &\equiv \mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{F})@\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{C}) \\
&\equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{F})@\mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{C}) \\
&\equiv \mathcal{E}_\varphi([\mathbf{B}/X](\mathbf{FC}))
\end{aligned}$$

Thus we have considered all cases and proven the assertion. □

Lemma 1.3.27 (Substitution-Value Lemma) *Let $(\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation. For any types α and β, variables X_β, and formulae $\mathbf{A} \in wff_\alpha(\Sigma)$ and $\mathbf{B} \in wff_\beta(\Sigma)$, we have $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$.*

Proof: By Lemma 1.3.25, there is an **SK**-combinatory formula $\mathbf{A}'$ such that $\mathbf{A}'$ β-reduces to $\mathbf{A}$. It follows by induction on the number of β-reductions from $\mathbf{A}'$ to $\mathbf{A}$ that $[\mathbf{B}/X]\mathbf{A}'$ β-reduces to $[\mathbf{B}/X]\mathbf{A}$. By Lemma 1.3.26 and the fact that $\mathcal{E}$ respects β-equality (cf. Remark 1.3.18), we have $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A}) \equiv \mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A}') \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A}') \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$. □

Example 1.3.28 (Singleton Evaluation) The singleton applicative structure (cf. Example 1.3.7) is a Σ-evaluation if for any assignment φ and formula $\mathbf{A}$ we take $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathsf{a}$, where a is the (unique) member of $\mathcal{D}_\alpha$. Note that in this Σ-evaluation $\mathcal{E}(\lambda X.X) \equiv \mathcal{E}_\varphi(\lambda X.Y)$ for any assignment φ.

For a detailed discussion on the closure conditions needed for the domains for function types to be rich enough for evaluation functions to exist, we refer the reader to [And72a, And73].

Note that the applicative term structure $wff(\Sigma)$ from Example 1.3.8 cannot be made into a Σ-evaluation by providing an evaluation function. To see this, suppose $\mathcal{E}$ is an evaluation function for $wff(\Sigma)$ and $\mathbf{F} := \mathcal{E}(\lambda X_\alpha.X) \in wff_{\alpha\to\alpha}(\Sigma)$. Since $\mathcal{E}$ is assumed to be an evaluation function, we must have $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi((\lambda X_\alpha.X)\mathbf{A}) \equiv \mathbf{F}@\mathbf{A} \equiv \mathbf{FA}$ for every $\mathbf{A} \in wff_\alpha(\Sigma)$. In particular, for any constant $a_\alpha \in \Sigma_\alpha$, we must have $\mathbf{F}a \equiv \mathcal{E}_\varphi(a) \equiv \mathcal{E}((\lambda X_\alpha.X)a) \equiv \mathcal{E}(\lambda X_\alpha.X)@\mathcal{E}(a) \equiv \mathbf{F}(\mathbf{F}a)$. But clearly $\mathbf{F}a \not\equiv \mathbf{F}(\mathbf{F}a)$ no matter what $\mathbf{F} \in wff_{\alpha\to\alpha}(\Sigma)$ we choose. In particular, the "obvious" choice of $\mathcal{E}(\lambda X_\alpha.X) \equiv (\lambda X_\alpha.X)$ does not work. This example suggests that we need to consider β-convertible terms equal before we can obtain a term evaluation (cf. Definition 1.3.35).

Definition 1.3.29 (Σ-Evaluation Congruences)
A *congruence* on a Σ-evaluation $\mathcal{J} \equiv (\mathcal{D}, @, \mathcal{E})$ is a congruence on the underlying applicative structure $(\mathcal{D}, @)$. Given any two variable assignments into $(\mathcal{D}, @)$, we will use the notation $\varphi \sim \psi$ to indicate that $\varphi(X) \sim \psi(X)$ for every variable X.

A typed equivalence relation was defined to be a congruence if it respects application. In order to form a quotient of a Σ-evaluation, we must be able to define an evaluation function $\mathcal{E}^\sim$ on the quotient structure. But $\mathcal{E}^\sim$ interprets all terms, including λ-abstractions. It is not obvious one can find a well-defined $\mathcal{E}^\sim$ that is really an evaluation function. In fact, the property one needs in order to show $\mathcal{E}^{\sim}$ will be a well-defined evaluation function is $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\psi(\mathbf{A})$ for every $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$ and assignments φ and ψ with $\varphi \sim \psi$. One can show this by an easy induction on the term $\mathbf{A}$ if the congruence $\sim$ is functional. However, without the assumption that $\sim$ is functional, this direct proof will fail when $\mathbf{A}$ is a λ-abstraction. Again, we avoid this problem using combinators.

Lemma 1.3.30 *Let $\mathcal{J} \equiv (\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation, $\sim$ a congruence on $\mathcal{J}$, and φ and ψ assignments into $\mathcal{J}$ with $\varphi \sim \psi$. For every* **SK***-combinatory formula* $\mathbf{A}$*, we have $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\psi(\mathbf{A})$.*

Proof: The proof is by induction on the **SK**-combinatory formula $\mathbf{A}$. If $\mathbf{A}$ is a variable X, we have $\mathcal{E}_\varphi(X) \equiv \varphi(X) \sim \psi(X) \equiv \mathcal{E}_\psi(X)$. If $\mathbf{A}$ is closed (e.g., a constant in Σ or a combinator), then $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\psi(\mathbf{A})$, so certainly $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\psi(\mathbf{A})$. Finally, if $\mathbf{A}$ is an application of two **SK**-combinatory formulae $\mathbf{F}$ and $\mathbf{B}$, then by the inductive hypothesis we have $\mathcal{E}_\varphi(\mathbf{F}) \sim \mathcal{E}_\psi(\mathbf{F})$ and $\mathcal{E}_\varphi(\mathbf{B}) \sim \mathcal{E}_\psi(\mathbf{B})$. Since $\sim$ respects application, $\mathcal{E}_\varphi(\mathbf{FB}) \equiv \mathcal{E}_\varphi(\mathbf{F})@\mathcal{E}_\varphi(\mathbf{B}) \sim \mathcal{E}_\psi(\mathbf{F})@\mathcal{E}_\psi(\mathbf{B}) \equiv \mathcal{E}_\psi(\mathbf{FB})$. □

We can use this result to easily show the same property holds for all formulae.

Lemma 1.3.31 *Let $\mathcal{J} \equiv (\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation, φ and ψ assignments into $\mathcal{J}$ with $\varphi \sim \psi$, and $\sim$ a congruence on $\mathcal{J}$. For every formula $\mathbf{A}$, we have $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\psi(\mathbf{A})$.*

Proof: Let $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$ for some type α. By Lemma 1.3.25 there is an **SK**-combinatory formula $\mathbf{A}'$ that β-reduces to $\mathbf{A}$. By Remark 1.3.18 and Lemma 1.3.30, we have $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{A}') \sim \mathcal{E}_\psi(\mathbf{A}') \equiv \mathcal{E}_\psi(\mathbf{A})$. □

Remark 1.3.32 (Correspondence with Logical Relations) Lemma 1.3.31 is essentially an instance of the "Basic Lemma" for logical relations (Lemma 8.2.5 in [Mit96]). In fact, $\sim$ is functional, iff $\sim$ is a logical relation over the applicative structure. If $\sim$ is not functional, it still satisfies this "Basic Lemma" property, which makes it a pre-logical relation in the sense of [HS99].

Definition 1.3.33 (Quotient Σ-Evaluation) Let $\mathcal{J} \equiv (\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation, $\sim$ a congruence on $\mathcal{J}$ and let $(\mathcal{D}^{\sim}, @^{\sim})$ be the quotient applicative structure of $(\mathcal{D}, @)$ with respect to $\sim$.

For each $\mathsf{A} \in \mathcal{D}^{\sim}_\alpha$, we choose a representative $\mathsf{A}^* \in \mathsf{A}$. So, $[\![\mathsf{A}^*]\!]_\sim \equiv \mathsf{A}$. Note that $[\![\mathsf{a}]\!]^*_\sim \sim \mathsf{a}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$. For any assignment φ into $\mathcal{J}/_\sim$, let φ^* be the assignment into $\mathcal{J}$ given by $\varphi^*(X) := \varphi(X)^*$. Note that $\varphi \equiv \pi_\sim \circ \varphi^*$. So we can define $\mathcal{E}^{\sim}_\varphi$ as $\pi_\sim \circ \mathcal{E}_{\varphi^*}$, and call $\mathcal{J}/_\sim := (\mathcal{D}^{\sim}, @^{\sim}, \mathcal{E}^{\sim})$ the *quotient Σ-evaluation* of $\mathcal{J}$ modulo $\sim$. (By Lemma 1.3.31, the definition of $\mathcal{E}^{\sim}$ does not depend on the choice of representatives.)

This definition is justified by the following theorem.

Theorem 1.3.34 (Quotient Σ-Evaluation Theorem) *If $\mathcal{J}$ is a Σ-evaluation and $\sim$ is a congruence on $\mathcal{J}$, then $\mathcal{J}/_\sim$ is a Σ-evaluation.*

Proof: We prove that $\mathcal{E}^{\sim}$ is an evaluation function by verifying the conditions in Definition 1.3.17. For any assignment φ into the quotient applicative structure, let φ^* be the assignment with $\varphi \equiv \pi_\sim \circ \varphi^*$ as in Definition 1.3.33.

1. $\mathcal{E}^{\sim}_\varphi\big|_{\mathcal{V}} \equiv (\pi_\sim \circ \mathcal{E}_{\varphi^*})\big|_{\mathcal{V}} \equiv \pi_\sim \circ \mathcal{E}_{\varphi^*}\big|_{\mathcal{V}} \equiv \pi_\sim \circ \varphi^* \equiv \varphi$.

2. Since $\pi_\sim$ is a homomorphism we have

$$\begin{aligned}\mathcal{E}^\sim_\varphi(\mathbf{FA}) &\equiv \pi_\sim(\mathcal{E}_{\varphi^*}(\mathbf{FA}))\\ &\equiv \pi_\sim(\mathcal{E}_{\varphi^*}(\mathbf{F})@\mathcal{E}_{\varphi^*}(\mathbf{A}))\\ &\equiv \pi_\sim(\mathcal{E}_{\varphi^*}(\mathbf{F}))@^\sim\pi_\sim(\mathcal{E}_{\varphi^*}(\mathbf{A}))\\ &\equiv \mathcal{E}^\sim_\varphi(\mathbf{F})@^\sim\mathcal{E}^\sim_\varphi(\mathbf{A})\end{aligned}$$

3. $\mathcal{E}^\sim_\varphi(\mathbf{A}) \equiv [\![\mathcal{E}_{\varphi^*}(\mathbf{A})]\!]_\sim \equiv [\![\mathcal{E}_{\psi^*}(\mathbf{A})]\!]_\sim \equiv \mathcal{E}^\sim_\psi(\mathbf{A})$, if φ and ψ coincide on $free(\mathbf{A})$, since this entails that φ^* and ψ^* do too (as we have chosen particular representatives for each equivalence class).

4. $\mathcal{E}^\sim_\varphi(\mathbf{A}) \equiv [\![\mathcal{E}_{\varphi^*}(\mathbf{A})]\!]_\sim \equiv [\![\mathcal{E}_{\varphi^*}(\mathbf{A}{\downarrow_\beta})]\!]_\sim \equiv \mathcal{E}^\sim_\varphi(\mathbf{A}{\downarrow_\beta})$. □

Definition 1.3.35 (Term Evaluations for Σ) Let $\mathit{cwff}(\Sigma)\big\downarrow_\beta$ be the collection of closed well-formed formulae in β-normal form and $\mathbf{A}@^\beta\mathbf{B}$ be $(\mathbf{AB})\big\downarrow_\beta$. For the definition of an evaluation function let φ be an assignment into $\mathit{cwff}(\Sigma)\big\downarrow_\beta$. Note that $\sigma := \varphi\big|_{free(\mathbf{A})}$ is a substitution, since $free(\mathbf{A})$ is finite. Thus we can choose $\mathcal{E}^\beta_\varphi(\mathbf{A}) := \sigma(\mathbf{A})\big\downarrow_\beta$. We call $\mathcal{TE}(\Sigma)^\beta := (\mathit{cwff}(\Sigma)\big\downarrow_\beta, @^\beta, \mathcal{E}^\beta)$ the *β-term evaluation* for Σ.

Analogously, we can define $\mathcal{TE}(\Sigma)^{\beta\eta} := (\mathit{cwff}(\Sigma)\big\downarrow_{\beta\eta}, @^{\beta\eta}, \mathcal{E}^{\beta\eta})$ the *$\beta\eta$-term evaluation* for Σ.

The name *term evaluation* in the previous definition is justified by the following lemma.

Lemma 1.3.36 *$\mathcal{TE}(\Sigma)^\beta$ is a Σ-evaluation and $\mathcal{TE}(\Sigma)^{\beta\eta}$ is a functional Σ-evaluation.*

Proof: The fact that $(\mathit{cwff}(\Sigma)\big\downarrow_\beta, @^\beta)$ is an applicative structure is immediate: For each type α, $\mathit{cwff}_\alpha(\Sigma)\big\downarrow_\beta$ is non-empty (by Remark 1.3.16) and

$$@^\beta \colon \mathit{cwff}_{\alpha\to\beta}(\Sigma)\big\downarrow_\beta \times \mathit{cwff}_\alpha(\Sigma)\big\downarrow_\beta \longrightarrow \mathit{cwff}_\beta(\Sigma)\big\downarrow_\beta$$

We next check that $\mathcal{E}^\beta$ is an evaluation function.

1. $\mathcal{E}^\beta_\varphi(X) \equiv \varphi\big|_{free(X)}(X) \equiv \varphi(X)$.

2. $\mathcal{E}^\beta_\varphi$ respects application since $\sigma(\mathbf{FA})\big\downarrow_\beta \equiv \Big(\sigma(\mathbf{F})\big\downarrow_\beta\ \sigma(\mathbf{A})\big\downarrow_\beta\Big)\Big\downarrow_\beta$ where $\sigma \equiv \varphi\big|_{free(\mathbf{A})}$.

3. $\mathcal{E}^\beta_\varphi(\mathbf{A}) \equiv \Big(\varphi\big|_{free(\mathbf{A})}(\mathbf{A})\Big)\Big\downarrow_\beta \equiv \Big(\varphi'\big|_{free(\mathbf{A})}(\mathbf{A})\Big)\Big\downarrow_\beta \equiv \mathcal{E}^\beta_{\varphi'}(\mathbf{A})$ whenever φ and φ' coincide on $free(\mathbf{A})$.

4. $\mathcal{E}^\beta_\varphi(\mathbf{A}) \equiv \sigma(\mathbf{A})\big\downarrow_\beta \equiv \sigma(\mathbf{A}{\downarrow_\beta})\big\downarrow_\beta \equiv \mathcal{E}^\beta_\varphi(\mathbf{A}{\downarrow_\beta})$ where $\sigma \equiv \varphi\big|_{free(\mathbf{A})}$.

A similar argument shows that $\mathcal{TE}(\Sigma)^{\beta\eta}$ is a Σ-evaluation. Also, one can show $\mathcal{TE}(\Sigma)^{\beta\eta}$ is functional using an argument similar to Lemma 1.3.14 since Σ is infinite at all types by Remark 1.3.16. (Alternatively, one can simply apply Lemma 1.3.14 and Theorem 1.3.13 to note that the applicative structure $\mathit{cwff}(\Sigma)/_{\equiv_{\beta\eta}}$ is functional. The applicative structure $\mathit{cwff}(\Sigma)/_{\equiv_{\beta\eta}}$ is isomorphic to the applicative structure $(\mathit{cwff}(\Sigma)\big\downarrow_{\beta\eta}, @^{\beta\eta})$. One can easily show that functionality is preserved under isomorphism.) □

Remark 1.3.37 Note that $\mathcal{TE}(\Sigma)^\beta$ is not a functional Σ-evaluation since, for instance,

$$(\lambda X_\gamma . Y_{\gamma\to\delta} X) @^\beta \mathbf{C}_\gamma \equiv Y @^\beta \mathbf{C}$$

for all $\mathbf{C}$ in $\mathcal{TE}_\gamma(\Sigma)^\beta$ but $\lambda X . Y X \not\equiv Y$.

Remark 1.3.38 One can show that an evaluation function $\mathcal{E}$ for an applicative structure $(\mathcal{D}, @)$ is uniquely determined by its values $\mathcal{E}(c)$ on the constants $c \in \Sigma$ and its values $\mathcal{E}(\mathbf{S})$ and $\mathcal{E}(\mathbf{K})$ on the combinators $\mathbf{S}$ and $\mathbf{K}$. When the applicative structure is functional, even the values of each $\mathcal{E}(\mathbf{S})$ and $\mathcal{E}(\mathbf{K})$ are determined, so that $\mathcal{E}$ is uniquely determined by its values $\mathcal{E}(c)$ for $c \in \Sigma$.

Definition 1.3.39 (Homomorphism on Σ-Evaluations)
Let $\mathcal{J}^1 := (\mathcal{D}^1, @^1, \mathcal{E}^1)$ and $\mathcal{J}^2 := (\mathcal{D}^2, @^2, \mathcal{E}^2)$ be Σ-evaluations. A Σ-*homomorphism* is a typed function $\kappa: \mathcal{D}^1 \longrightarrow \mathcal{D}^2$ such that κ is a homomorphism from the applicative structure $(\mathcal{D}^1, @^1)$ to the applicative structure $(\mathcal{D}^2, @^2)$ and $\kappa\left(\mathcal{E}^1_\varphi(\mathbf{A})\right) \equiv \mathcal{E}^2_{\kappa\circ\varphi}(\mathbf{A})$ for every $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$ and assignment φ for $\mathcal{J}^1$.

1.3.3 Σ-Models

The semantic notions so far are independent of the set of base types. Now, we specialize these to obtain a notion of models by requiring specialized behavior on the type o of truth values. For this we use the notion of a Σ-valuation which gives a truth-value interpretation to the domain $\mathcal{D}_o$ of a Σ-evaluation consistent with the intuitive interpretations of the logical constants. Since models are semantic entities that are constructed primarily to make a statement about the truth or falsity of a formula, the requirement that there exists a Σ-valuation is perhaps the most general condition under which one wants to speak of a model. Thus we will define our most general notion of semantics as Σ-evaluations that have Σ-valuations.

Definition 1.3.40 Fix two values $\mathtt{T} \not\equiv \mathtt{F}$. Let $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation and $\upsilon: \mathcal{D}_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ be a (total) function. We define several properties that characterize logical operators with respect to υ in the table shown in Figure 1.2.

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

Figure 1.2: Logical Properties in Σ-Models

Definition 1.3.41 (Σ-Model)
Let $\mathcal{J} := (\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation. A function $\upsilon: \mathcal{D}_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ is called a Σ-*valuation* for $\mathcal{J}$ if $\mathfrak{L}_\neg(\mathcal{E}(\neg))$ and $\mathfrak{L}_\vee(\mathcal{E}(\vee))$ hold, and for every type α $\mathfrak{L}^\alpha_\forall(\mathcal{E}(\Pi^\alpha))$ holds. In this case, $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ is called a Σ-*model*.

For the case of (the optional) primitive equality, i.e. when $=^{\alpha} \in \Sigma_{\alpha\to\alpha\to o}$ for all types α, we say $\mathcal{M}$ is a Σ-*model with primitive equality* if $\mathfrak{L}^{\alpha}_{=}(\mathcal{E}(=^{\alpha}))$ holds for every type α.

We say φ is an assignment into $\mathcal{M}$ if it is an assignment into the underlying applicative structure $(\mathcal{D}, @)$. Furthermore, φ *satisfies* a formula $\mathbf{A} \in \mathit{wff}_o(\Sigma)$ in $\mathcal{M}$ (we write $\mathcal{M} \models_{\varphi} \mathbf{A}$) if $\upsilon(\mathcal{E}_{\varphi}(\mathbf{A})) \equiv \mathtt{T}$. 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 the reference to the assignment and use the notation $\mathcal{M} \models \mathbf{A}$. Finally, we say that $\mathcal{M}$ is a 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 $\mathbf{A} \in \Phi$.

A Σ-model $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ is called *functional* [*full*, *standard*] if the applicative structure $(\mathcal{D}, @)$ is *functional* [*full*, *standard*]. 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 a Σ-model over a frame if $(\mathcal{D}, @)$ is a frame.

Remark 1.3.42 (Adding Primitive Equality) In the definition of Σ-model above, the addition of property $\mathfrak{L}^{\alpha}_{=}(\mathcal{E}(=^{\alpha}))$ addressing the case of primitive equality above has a purely practical motivation: calculi with a primitive treatment of equality, see for instance [Ben99a, Ben99b], may provide a more effective approach to equational reasoning in higher order logic than the exclusive use of Leibniz equality. Therefore we enrich our theory to automatically also address the situation where (always built-in) Leibniz equality and (optional) primitive equality are simultaneously present in the language. The generalization to primitive equality is less trivial than the generalization to other (optional) primitive logical connectives such as $\wedge$ or $\Rightarrow$. This is the main reason why we built primitive equality directly into our theory while we omit other logical primitives (cf. also Remarks 1.3.47 and 1.6.9).

Lemma 1.3.43 (Truth and Falsity in Σ-Models) *Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model and φ an assignment. Let $\mathbf{T}_o := \forall P_o\text{.}P \vee \neg P$ and $\mathbf{F}_o := \neg \mathbf{T}_o$. Then $\upsilon(\mathcal{E}_{\varphi}(\mathbf{T}_o)) \equiv \mathtt{T}$ and $\upsilon(\mathcal{E}_{\varphi}(\mathbf{F}_o)) \equiv \mathtt{F}$.*

Proof: Let P be a variable of type o. We have $\upsilon(\mathcal{E}_{\varphi}(\mathbf{T}_o)) \equiv \mathtt{T}$, iff $\upsilon(\mathcal{E}_{\varphi}(P \vee \neg P)) \equiv \mathtt{T}$ for every assignment φ. The properties of υ show that this statement is equivalent to $\upsilon(\varphi(P)) \equiv \mathtt{T}$ or $\upsilon(\varphi(P)) \equiv \mathtt{F}$, which is always true since υ maps into $\{\mathtt{T}, \mathtt{F}\}$. Note further that $\upsilon(\mathcal{E}_{\varphi}(\mathbf{F}_o)) \equiv \mathtt{F}$ since $\upsilon(\mathcal{E}_{\varphi}(\mathbf{T}_o)) \equiv \mathtt{T}$. □

Remark 1.3.44 Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model. By Lemma 1.3.43, $\mathcal{D}_o$ 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.

Remark 1.3.45 In contrast to the case of Henkin Models, Definition 1.3.41 only constrains the functional behavior of the values of the logical constants with respect to υ. This does not fully specify these values since

- $\mathcal{M}$ need not be functional,
- and there can be more than two truth values.

We will now introduce semantical properties called $\mathfrak{q}$, η, $\mathfrak{f}$, and $\mathfrak{b}$, which we will use to characterize different classes of Σ-models.

Definition 1.3.46 (Properties $\mathfrak{q}$, η, ξ, $\mathfrak{f}$ and $\mathfrak{b}$)
Given a 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 $\mathrm{q}^{\alpha} \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that $\mathfrak{L}^{\alpha}_{=}(\mathrm{q}^{\alpha})$ 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 $\mathcal{D}_o$ has at most two elements. By Lemma 1.3.44 we can assume without loss of generality that $\mathcal{D}_o \equiv \{\mathtt{T}, \mathtt{F}\}$, υ is the identity function, $\mathcal{E}_\varphi(\mathbf{T}_o) \equiv \mathtt{T}$ and $\mathcal{E}_\varphi(\mathbf{F}_o) \equiv \mathtt{F}$. (This is generally associated with Boolean extensionality.)

Remark 1.3.47 (Choice of Logical Constants) The work presented in this article is based on the choice of the primitive logical constants $\neg$, $\vee$, and Π^α. We have also introduced shorthand for formulas constructed using $\wedge$, $\Rightarrow$, $\Leftrightarrow$, and existential quantification. One can (easily; cf. Lemma 1.3.48) verify that in any Σ-model $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$, each of the properties $\mathfrak{L}_\wedge(\mathcal{E}(\lambda X_o Y_o \text{.} X \wedge Y))$, $\mathfrak{L}_\Rightarrow(\mathcal{E}(\lambda X_o Y_o \text{.} X \Rightarrow Y))$, $\mathfrak{L}_\Leftrightarrow(\mathcal{E}(\lambda X_o Y_o \text{.} X \Leftrightarrow Y))$ and $\mathfrak{L}_\exists^\alpha(\mathcal{E}(\lambda P_{\alpha\to o} \text{.} \exists X_\alpha \text{.} PX))$ (for each type α) hold with respect to υ. In this sense, our choice of logical constants and shorthand for other logical constants is sufficient. However, Leibniz equality $\mathbf{Q}^\alpha$ will only satisfy $\mathfrak{L}_=^\alpha(\mathcal{E}(\mathbf{Q}^\alpha))$ for each type α iff the model satisfies property $\mathfrak{q}$ (cf. Remark 1.3.52 and Theorem 1.3.63).

On the other hand, in the absence of extensionality, one can gain some (limited) expressive power by including extra logical constants such as $\wedge$ in the signature. This is the case since there may be several objects in $\mathsf{c} \in \mathcal{D}_{o\to o\to o}$ such that $\mathfrak{L}_\wedge(\mathsf{c})$ holds. So, one could have a Σ-model $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ (where $\wedge$ is also in Σ) such that $\mathfrak{L}_\wedge(\mathcal{E}(\wedge))$ holds, but $\mathcal{E}(\wedge) \not\equiv \mathcal{E}(\lambda X_o Y_o \text{.} \neg(\neg X \vee \neg Y))$. (We will not investigate this possibility here.)

Our choice of logical constants differs from Andrews' choice [And02] who considers primitive equality as the only logical primitive from which all other logical operators are defined using the definitions in Figure 1.3. For the sake of clarity, we write $\mathbf{q}^\alpha$ for $=^\alpha$ when $=^\alpha$ is not being written in infix notation. For Henkin models, the definitions in Figure 1.3 are appropriate. However,

$\mathbf{T}_o$	$:=$	$\mathbf{q}^o =^{o\to o\to o} \mathbf{q}^o$
$\mathbf{F}_o$	$:=$	$(\lambda X_o \text{.} \mathbf{T}_o) =^{o\to o} (\lambda X_o \text{.} X)$
$\neg_{o\to o}$	$:=$	$\mathbf{q}^o \mathbf{F}_o$
Π^α	$:=$	$\mathbf{q}^{\alpha\to o}(\lambda X_\alpha \text{.} \mathbf{T}_o)$
$\wedge_{o\to o\to o}$	$:=$	$\lambda X_o Y_o \text{.} (\lambda G_{o\to o\to o} \text{.} G\mathbf{T}_o\mathbf{T}_o) =^{o\to o\to o} (\lambda G_{o\to o\to o} \text{.} GXY)$
$\Rightarrow_{o\to o\to o}$	$:=$	$\lambda X_o Y_o \text{.} (X =^{o\to o\to o} (X \wedge Y))$
$\vee_{o\to o\to o}$	$:=$	$\lambda X_o Y_o \text{.} \neg(\neg X \wedge \neg Y)$
Σ^α	$:=$	$\lambda P_{\alpha\to o} \text{.} (\neg \Pi^\alpha \lambda X_\alpha \text{.} \neg(PX))$

Figure 1.3: A definition of logical constants from equality in Henkin Models

without extensionality, the situation is quite different. Suppose $\mathcal{J} \equiv (\mathcal{D}, @, \mathcal{E})$ is a Σ-evaluation where $=^\alpha \in \Sigma$ for every type α. Let $\upsilon : \mathcal{D}_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ be a function such that $\mathfrak{L}_=^\alpha(\mathcal{E}(=^\alpha))$ holds for each type α. The fact that $\upsilon(\mathcal{E}(\mathbf{T}_o)) \equiv \mathtt{T}$ follows directly from $\mathfrak{L}_=^{o\to o\to o}(\mathcal{E}(=^{o\to o\to o}))$ and reflexivity of (metalevel) equality. Unfortunately, this is the last definition which is clearly appropriate without further assumptions. So long as $\mathcal{D}_o$ has more than one element, one can show $\upsilon(\mathcal{E}(\mathbf{F}_o)) \equiv \mathtt{F}$. So, let us explicitly assume $\mathcal{D}_o$ has more than one element, which is anyway met by Σ-models (cf. Remark 1.3.47). Next, we investigate whether $\mathfrak{L}_\neg(\mathcal{E}(\neg))$ holds. Let $\mathsf{a} \in \mathcal{D}_o$ be given. By $\mathfrak{L}_=^o(\mathcal{E}(=^o))$, we know $\upsilon(\mathcal{E}(=^o)@\mathcal{E}(\mathbf{F}_o)@\mathsf{a}) \equiv \mathtt{T}$ is equivalent to $\mathcal{E}(\mathbf{F}_o) \equiv \mathsf{a}$. So, if $\upsilon(\mathcal{E}(=^o)@\mathcal{E}(\mathbf{F}_o)@\mathsf{a}) \equiv \mathtt{T}$, then $\upsilon(\mathsf{a}) \equiv \upsilon(\mathcal{E}(\mathbf{F}_o)) \equiv \mathtt{F}$. For the converse, suppose $\upsilon(\mathsf{a}) \equiv \mathtt{F}$. This, in general, does not imply $\mathcal{E}(\mathbf{F}_o) \equiv \mathsf{a}$. However, if we assume a is the *unique* member of $\mathcal{D}_o$ such that $v(\mathsf{a}) \equiv \mathtt{F}$, then we can conclude $\mathcal{E}(\mathbf{F}_o) \equiv \mathsf{a}$. In particular, if $\mathcal{D}_o$ has only two elements, then v must be injective and we can conclude $\mathcal{E}(\mathbf{F}_o) \equiv \mathsf{a}$. So, Boolean extensionality is required to ensure that $\mathfrak{L}_\neg(\mathcal{E}(\neg))$ holds for this definition of $\neg$.

We now investigate whether $\mathfrak{L}_\forall^\alpha(\mathcal{E}(\Pi^\alpha))$ holds for Π^α defined as in Figure 1.3. Let $\mathsf{f} \in \mathcal{D}_{\alpha\to o}$ be given. Suppose $v(\mathcal{E}(=^{\alpha\to o})@\mathcal{E}(\lambda X_\alpha.\mathbf{T}_o)@\mathsf{f}) \equiv \mathtt{T}$. By $\mathfrak{L}_=^{\alpha\to o}(\mathcal{E}(=^{\alpha\to o}))$, we know $\mathcal{E}(\lambda X_\alpha.\mathbf{T}_o) \equiv \mathsf{f}$. This does guarantee $\mathcal{E}(\mathbf{T}_o) \equiv \mathsf{f}@\mathsf{a}$ and hence $v(\mathsf{f}@\mathsf{a}) \equiv \mathtt{T}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$. However, showing the converse requires that $\mathcal{M}$ is functional (i.e. strong functional extensionality is given). Suppose $v(\mathcal{E}(=^\alpha)@\mathcal{E}(\lambda X_\alpha.\mathbf{T}_o)@\mathsf{f}) \equiv \mathtt{F}$. We can conclude $\mathcal{E}(\lambda X_\alpha.\mathbf{T}_o) \not\equiv \mathsf{f}$, but this is of little value. If $\mathcal{J}$ is not functional, then these may be different representatives in $\mathcal{D}_{\alpha\to o}$ of the same function. If $\mathcal{J}$ is functional, there must be some $\mathsf{a} \in \mathcal{D}_\alpha$ such that $\mathcal{E}(\mathbf{T}_o) \not\equiv \mathsf{f}@\mathsf{a}$. However, this still does not imply $v(\mathsf{f}@\mathsf{a}) \equiv \mathtt{F}$. If $\mathcal{D}_o$ has only two elements, then the facts that $\mathcal{E}(\mathbf{T}_o) \not\equiv \mathsf{f}@\mathsf{a}$ and $\mathcal{E}(\mathbf{T}_o) \not\equiv \mathcal{E}(\mathbf{F}_o)$ imply $\mathcal{E}(\mathbf{F}_o) \equiv \mathsf{f}@\mathsf{a}$, hence $v(\mathsf{f}@\mathsf{a}) \equiv \mathtt{F}$.

Similar observations apply to the other definitions in Figure 1.3. These definitions do show that at least $\mathbf{T}_o$ and $\mathbf{F}_o$ are definable from primitive equality (so long as $\mathcal{D}_o$ has at least two elements). Furthermore, if $\mathcal{D}_o$ has exactly two elements $\neg$ is definable from primitive equality. We conjecture that this is as much as one can define in terms of primitive equality without extensionality assumptions. That is, we conjecture that without assuming $\mathcal{D}_o$ has two elements, there may be no object $\mathsf{n} \in \mathcal{D}_{o\to o}$ such that $\mathfrak{L}_\neg(\mathsf{n})$ holds. Furthermore, we conjecture that without assuming functionality and that $\mathcal{D}_o$ has two elements, there may be no object $\mathsf{d} \in \mathcal{D}_{o\to o\to o}$ such that $\mathfrak{L}_\vee(\mathsf{d})$ holds, and there may be no object $\pi \in \mathcal{D}_{(\alpha\to o)\to o}$ such that $\mathfrak{L}_\forall^\alpha(\pi)$ holds.

The next lemma formally verifies that $\mathfrak{L}_\Leftrightarrow(\mathcal{E}(\lambda X_o Y_o.X \Leftrightarrow Y))$ holds with respect to the valuation of a Σ-model, as indicated in the remark above.

Lemma 1.3.48 (Equivalence) *Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, v)$ be a Σ-model, φ an assignment into $\mathcal{M}$, and $\mathbf{A}, \mathbf{B} \in \mathit{wff}_o(\Sigma)$. $v(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) \equiv \mathtt{T}$ iff $v(\mathcal{E}_\varphi(\mathbf{A})) \equiv v(\mathcal{E}_\varphi(\mathbf{B}))$.*

Proof: Suppose $v(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) \equiv \mathtt{T}$. This implies $v(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) \equiv \mathtt{T}$ and $v(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) \equiv \mathtt{T}$. If $v(\mathcal{E}_\varphi(\mathbf{A})) \equiv \mathtt{T}$, then $v(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) \equiv \mathtt{T}$ implies $v(\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{T}$, so $v(\mathcal{E}_\varphi(\mathbf{A})) \equiv \mathtt{T} \equiv v(\mathcal{E}_\varphi(\mathbf{B}))$. If $v(\mathcal{E}_\varphi(\mathbf{A})) \equiv \mathtt{F}$, then $v(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) \equiv \mathtt{T}$ implies $v(\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{F}$, so $v(\mathcal{E}_\varphi(\mathbf{A})) \equiv \mathtt{F} \equiv v(\mathcal{E}_\varphi(\mathbf{B}))$. Since these are the only two possible values for $v(\mathcal{E}_\varphi(\mathbf{A}))$, we have $v(\mathcal{E}_\varphi(\mathbf{A})) \equiv v(\mathcal{E}_\varphi(\mathbf{B}))$.

Suppose $v(\mathcal{E}_\varphi(\mathbf{A})) \equiv v(\mathcal{E}_\varphi(\mathbf{B}))$. Either $v(\mathcal{E}_\varphi(\mathbf{A})) \equiv v(\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{T}$ or $v(\mathcal{E}_\varphi(\mathbf{A})) \equiv v(\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{F}$. An easy consideration of both cases verifies $v(\mathcal{E}_\varphi(\neg\mathbf{A} \vee \mathbf{B})) \equiv \mathtt{T}$ and $v(\mathcal{E}_\varphi(\neg\mathbf{B} \vee \mathbf{A})) \equiv \mathtt{T}$. Hence, $v(\mathcal{E}_\varphi(\mathbf{A} \Leftrightarrow \mathbf{B})) \equiv \mathtt{T}$. □

Definition 1.3.49 (Higher Order Model Classes) We will denote the set of Σ-models that satisfy property $\mathfrak{q}$ by $\mathfrak{M}_\beta$,[5] and we will use subclasses of $\mathfrak{M}_\beta$ depending on the validity of the properties η, ξ, $\mathfrak{f}$, and $\mathfrak{b}$. We obtain the specialized classes of Σ-models $\mathfrak{M}_{\beta\eta}$, $\mathfrak{M}_{\beta\xi}$, $\mathfrak{M}_{\beta\mathfrak{f}}$, $\mathfrak{M}_{\beta\mathfrak{b}}$, $\mathfrak{M}_{\beta\eta\mathfrak{b}}$, $\mathfrak{M}_{\beta\xi\mathfrak{b}}$, and $\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$ by requiring that the properties specified in the index are valid.

If primitive equality is in the signature, i.e. if $=^\alpha \in \Sigma_{\alpha\to\alpha\to o}$, then we require the models to be Σ-models with primitive equality. Note that in this case property $\mathfrak{q}$ is automatically ensured.

We can group these eight classes in two dimensions as in Figure 1.4 based on the "amount of extensionality" required.

[5]The only Σ-models we need to consider which do not satisfy property $\mathfrak{q}$ are term models. It will turn out (cf. Theorem 1.3.62) that we can obtain a model satisfying property $\mathfrak{q}$ from a model that does not by taking a quotient. However, this may not preserve properties ξ or $\mathfrak{f}$.

functional

		none	weak (η)	weak (ξ)	strong ($\mathfrak{f}$)
Boolean	none	$\mathfrak{M}_{\beta}$	$\mathfrak{M}_{\beta\eta}$	$\mathfrak{M}_{\beta\xi}$	$\mathfrak{M}_{\beta\mathfrak{f}}$
	$\mathfrak{b}$	$\mathfrak{M}_{\beta\mathfrak{b}}$	$\mathfrak{M}_{\beta\eta\mathfrak{b}}$	$\mathfrak{M}_{\beta\xi\mathfrak{b}}$	$\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$

Figure 1.4: Extensional Model Classes

Definition 1.3.50 (Σ-Henkin Models) A Σ-*Henkin model* is a model $\mathcal{M}$ over a frame with $\mathcal{M} \in \mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$. We denote the class of all Σ-Henkin models by $\mathfrak{H}$. (Such models are called *general models* in [And72a] and [And02]. We avoid this terminology here since we consider models which are more general than these.)

Definition 1.3.51 (Σ-Standard Models) A Σ-*standard model* is a Σ-Henkin model that is also full (i.e., a model $\mathcal{M} \in \mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$ over a standard frame). The class of all Σ-standard models is denoted by $\mathfrak{ST}$.

Remark 1.3.52 (Property $\mathfrak{q}$) The purpose of property $\mathfrak{q}$ is to ensure that for all types α there is an object q^α in $\mathcal{D}_{\alpha\to\alpha\to o}$ representing meta equality for the domain $\mathcal{D}_\alpha$. This ensures the existence of objects representing unit sets $\{\mathsf{a}\}$ for each $\mathsf{a} \in \mathcal{D}_\alpha$ in the domains $\mathcal{D}_{\alpha\to o}$, which in turn makes Leibniz equality the intended equality relation. This is because membership in these unit sets can be used as an appropriately strong criterion to distinguish between different elements of $\mathcal{D}_\alpha$. This aspect is discussed in detail by Peter Andrews in [And72a]. He notes that Leon Henkin unintentionally introduced in [Hen50] a class of models which need not satisfy property $\mathfrak{q}$ instead of the class of Henkin models in the sense above. As Andrews shows, a consequence is that such a model may fail to satisfy the principle of strong functional extensionality (cf. Definition 1.4.5) given by the formula

$$\forall F_{\iota\to\iota}.\forall G_{\iota\to\iota}.(\forall X_\iota.FX \doteq^\iota GX) \Rightarrow F \doteq^{\iota\to\iota} G$$

even though the model (as a model over a frame) is functional. Andrews fixed this problem by introducing property $\mathfrak{q}$. Here, we have followed this by requiring property $\mathfrak{q}$ in all our model classes $\mathfrak{M}_*$.

Now let us extend the notion of a quotient evaluation to Σ-models.

Definition 1.3.53 (Σ-Model Congruences)
A *congruence* on a Σ-model $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ is a congruence on the underlying Σ-evaluation $(\mathcal{D}, @, \mathcal{E})$ such that $\upsilon(\mathsf{a}) \equiv \upsilon(\mathsf{b})$ for all $\mathsf{a}, \mathsf{b} \in \mathcal{D}_o$ with $\mathsf{a} \sim \mathsf{b}$.

Definition 1.3.54 (Quotient Σ-Model) Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, $\sim$ be a congruence on $\mathcal{M}$, and $(\mathcal{D}^\sim, @^\sim, \mathcal{E}^\sim)$ be the quotient Σ-evaluation of $(\mathcal{D}, @, \mathcal{E})$ with respect to $\sim$ (cf. Definition 1.3.33). Using the notation for representatives $\mathsf{A}^* \in \mathsf{A}$ for $\mathsf{A} \in \mathcal{D}^\sim_\alpha$ as in Definition 1.3.33, we define $\upsilon^\sim : \mathcal{D}^\sim_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ by $\upsilon^\sim(\mathsf{A}) := \upsilon(\mathsf{A}^*)$ for every $\mathsf{A} \in \mathcal{D}^\sim_o$. (Since $\upsilon(\mathsf{a}) \equiv \upsilon(\mathsf{b})$ whenever $\mathsf{a} \sim \mathsf{b}$ in $\mathcal{D}_o$, this definition of $\upsilon^\sim$ does not depend on the choice of representatives and $\upsilon^\sim([\![\mathsf{a}]\!]_\sim) \equiv \upsilon(\mathsf{a})$ for every $\mathsf{a} \in \mathcal{D}_o$.) We call $\mathcal{M}/_\sim := (\mathcal{D}^\sim, @^\sim, \mathcal{E}^\sim, \upsilon^\sim)$ the *quotient* Σ-model of $\mathcal{M}$ with respect to $\sim$.

Theorem 1.3.55 (Quotient Σ-Model Theorem) *Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model and $\sim$ be a congruence on $\mathcal{M}$. The quotient $\mathcal{M}/_{\sim}$ is a Σ-model.*

Furthermore, if for every type α, $=^\alpha \in \Sigma_\alpha$ and we have $\upsilon(\mathcal{E}(=^\alpha)@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$ iff $\mathsf{a} \sim \mathsf{b}$ for every $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$, then $\mathcal{M}/_{\sim}$ is a Σ-model with primitive equality.

Proof: We check the conditions of Definition 1.3.41, again using the A^* notation for representatives. To check condition $\mathfrak{L}_\neg(\mathcal{E}^\sim(\neg))$ for $\upsilon^\sim$, for all $\mathsf{A} \in \mathcal{D}_o^\sim$ we need to show that $\upsilon^\sim(\mathcal{E}^\sim(\neg)@^\sim\mathsf{A}) \equiv \mathtt{T}$ iff $\upsilon^\sim(\mathsf{A}) \equiv \mathtt{F}$. Let $\mathsf{A} \in \mathcal{D}_o^\sim$ be given. Since $\mathcal{M}$ is a Σ-model we have $\upsilon(\mathcal{E}(\neg)@\mathsf{A}^*) \equiv \mathtt{T}$ iff $\upsilon(\mathsf{A}^*) \equiv \mathtt{F}$. Since $[\![\mathsf{A}^*]\!]_\sim \equiv \mathsf{A}$ and $[\![\mathcal{E}(\neg)@\mathsf{A}^*]\!]_\sim \equiv \mathcal{E}^\sim(\neg)@^\sim\mathsf{A}$, we have $\upsilon^\sim(\mathcal{E}^\sim(\neg)@^\sim\mathsf{A}) \equiv \mathtt{T}$ iff $\upsilon^\sim(\mathsf{A}) \equiv \mathtt{F}$. Checking condition $\mathfrak{L}_\vee(\mathcal{E}^\sim(\vee))$ for $\upsilon^\sim$ is analogous.

To check condition $\mathfrak{L}_\forall^\alpha(\mathcal{E}^\sim(\Pi^\alpha))$ for $\upsilon^\sim$, suppose we have $\mathsf{G} \in \mathcal{D}_{\alpha\to o}^\sim$. For every $\mathsf{A} \in \mathcal{D}_\alpha^\sim$, $\upsilon^\sim(\mathsf{G}@^\sim\mathsf{A}) \equiv \upsilon(\mathsf{G}^*@\mathsf{A}^*)$. So, if $\upsilon^\sim(\mathsf{G}@^\sim\mathsf{A}) \equiv \mathtt{T}$ for every $\mathsf{A} \in \mathcal{D}_\alpha^\sim$, then $\upsilon(\mathsf{G}^*@\mathsf{a}) \equiv \upsilon(\mathsf{G}^*@[\![\mathsf{a}]\!]_\sim^*) \equiv \mathtt{T}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$, and we conclude $\upsilon(\mathcal{E}(\Pi^\alpha)@\mathsf{G}^*) \equiv \mathtt{T}$. Hence, $\upsilon^\sim(\mathcal{E}^\sim(\Pi^\alpha)@^\sim\mathsf{G}) \equiv \mathtt{T}$. Conversely, suppose $\upsilon^\sim(\mathcal{E}^\sim(\Pi^\alpha)@\mathsf{G}) \equiv \mathtt{T}$. Then $\upsilon(\mathcal{E}(\Pi^\alpha)@\mathsf{G}^*) \equiv \mathtt{T}$ and hence $\upsilon^\sim(\mathsf{G}@\mathsf{A}) \equiv \upsilon(\mathsf{G}^*@\mathsf{A}^*) \equiv \mathtt{T}$ for every $\mathsf{A} \in \mathcal{D}_\alpha^\sim$.

Suppose primitive equality is in the signature and $\upsilon(\mathcal{E}(=^\alpha)@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$ iff $\mathsf{a} \sim \mathsf{b}$ for every $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$. To verify $\mathfrak{L}_=^\alpha(\mathcal{E}^\sim(=^\alpha))$ holds for $\upsilon^\sim$, we simply note that $\upsilon^\sim(\mathcal{E}^\sim(=^\alpha)@^\sim\mathsf{A}@^\sim\mathsf{B}) \equiv \mathtt{T}$, iff $\upsilon(\mathcal{E}(=^\alpha)@\mathsf{A}^*@\mathsf{B}^*) \equiv \mathtt{T}$, iff $\mathsf{A}^* \sim \mathsf{B}^*$, iff $\mathsf{A} \equiv \mathsf{B}$. $\square$

We can define properties of a congruence analogous to those defined for models in Definition 1.3.46.

Definition 1.3.56 (Properties η, ξ, $\mathfrak{f}$ and $\mathfrak{b}$ for Congruences)
Given a Σ-model $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ and a congruence $\sim$ on $\mathcal{M}$, we say $\sim$ has *property*

η iff $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\varphi(\mathbf{A}{\downarrow_{\beta\eta}})$ for any type α, $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, and assignment φ.

ξ iff for all $\alpha, \beta \in \mathcal{T}$, $\mathbf{M}, \mathbf{N} \in \mathit{wff}_\beta(\Sigma)$, assignment φ, and variables X_α, $\mathcal{E}_\varphi(\lambda X_\alpha \boldsymbol{.} \mathbf{M}_\beta) \sim \mathcal{E}_\varphi(\lambda X_\alpha \boldsymbol{.} \mathbf{N}_\beta)$ whenever $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \sim \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N})$ for every $\mathsf{a} \in \mathcal{D}_\alpha$.

$\mathfrak{f}$ iff $\sim$ is functional.

$\mathfrak{b}$ iff $\mathcal{D}_o$ has at most two equivalence classes with respect to $\sim$. (By Remark 1.3.44 there are always at least two.)

Remark 1.3.57 It follows trivially from reflexivity of congruences that if a model satisfies property η, then any congruence on the model satisfies property η. Similarly, if a model has only two elements in $\mathcal{D}_o$, then $\mathcal{D}_o$ can have at most two equivalence classes with respect to any congruence $\sim$. So, if a model satisfies property $\mathfrak{b}$, then any congruence on the model satisfies property $\mathfrak{b}$. This is not true for properties ξ or $\mathfrak{f}$. For an example, we refer to the functional model (satisfying property $\mathfrak{f}$, hence property ξ) constructed by Andrews in [And72a]. Using the results we prove below, one can show Leibniz equality must induce a congruence failing to satisfy properties ξ and $\mathfrak{f}$ on this functional model.

Lemma 1.3.58 *Let $\mathcal{M}$ be a Σ-model, $\Phi \subseteq \mathit{cwff}_o(\Sigma)$, and $\sim$ be a congruence on $\mathcal{M}$. We have $\mathcal{M}/_{\sim} \models \Phi$ iff $\mathcal{M} \models \Phi$. Furthermore, if $* \in \{\eta, \xi, \mathfrak{f}, \mathfrak{b}\}$ and $\sim$ satisfies property $*$, then $\mathcal{M}/_{\sim}$ satisfies property $*$.*

Proof: Let $\mathbf{A}_o \in \Phi$. Since $\mathbf{A}$ is closed, $\mathcal{M} \models \mathbf{A}$, iff $\upsilon(\mathcal{E}(\mathbf{A})) \equiv \mathtt{T}$, iff $\upsilon^\sim(\mathcal{E}^\sim(\mathbf{A})) \equiv \mathtt{T}$, iff $\mathcal{M}/_{\sim} \models \mathbf{A}$. So, $\mathcal{M} \models \Phi$ iff $\mathcal{M}/_{\sim} \models \Phi$.

η: Suppose $\sim$ satisfies property η. Let $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, and an assignment φ into $\mathcal{M}/_\sim$ be given. Let φ^* be a corresponding assignment into $\mathcal{M}$ (cf. Definition 1.3.33). Since $\sim$ satisfies property η, we know $\mathcal{E}_{\varphi^*}(\mathbf{A}) \sim \mathcal{E}_{\varphi^*}(\mathbf{A}{\downarrow_{\beta\eta}})$. Taking equivalence classes, we have $\mathcal{E}^\sim_\varphi(\mathbf{A}) \equiv \mathcal{E}^\sim_\varphi(\mathbf{A}{\downarrow_{\beta\eta}})$.

ξ: Suppose $\sim$ satisfies property ξ. Let $\mathbf{M}, \mathbf{N} \in \mathit{wff}_\beta(\Sigma)$, a variable X_α and an assignment φ into $\mathcal{M}/_\sim$ be given. Again, let φ^* be a corresponding assignment into $\mathcal{M}$. Suppose $\mathcal{E}^\sim_{\varphi,[\mathsf{A}/X]}(\mathbf{M}) \equiv \mathcal{E}^\sim_{\varphi,[\mathsf{A}/X]}(\mathbf{N})$ for every $\mathsf{A} \in \mathcal{D}^\sim_\alpha$. This means $\mathcal{E}_{\varphi^*,[\mathsf{A}^*/X]}(\mathbf{M}) \sim \mathcal{E}_{\varphi^*,[\mathsf{A}^*/X]}(\mathbf{N})$ for every $\mathsf{A} \in \mathcal{D}^\sim_\alpha$. For any $\mathsf{a} \in \mathcal{D}_\alpha$, using Lemma 1.3.31, we know

$$\mathcal{E}_{\varphi^*,[\mathsf{a}/X]}(\mathbf{M}) \sim \mathcal{E}_{\varphi^*,[\mathsf{A}^*/X]}(\mathbf{M}) \sim \mathcal{E}_{\varphi^*,[\mathsf{A}^*/X]}(\mathbf{N}) \sim \mathcal{E}_{\varphi^*,[\mathsf{a}/X]}(\mathbf{N})$$

where $\mathsf{A} \in \mathcal{D}^\sim_\alpha$ is the equivalence class of a. Since $\sim$ satisfies property ξ, we know that $\mathcal{E}_{\varphi^*}(\lambda X.\mathbf{M}) \sim \mathcal{E}_{\varphi^*}(\lambda X.\mathbf{N})$. Taking equivalence classes, we see that $\mathcal{E}^\sim_\varphi(\lambda X.\mathbf{M}) \equiv \mathcal{E}^\sim_\varphi(\lambda X.\mathbf{N})$.

$\mathfrak{f}$: If $\sim$ is functional, we know $\mathcal{M}/_\sim$ is functional by Theorem 1.3.13.

$\mathfrak{b}$: Finally, if $\sim$ satisfies property $\mathfrak{b}$, then clearly $\mathcal{D}^\sim_o$ has only two elements. So, $\mathcal{M}/_\sim$ satisfies property $\mathfrak{b}$. □

Definition 1.3.59 (Congruence Relation $\dot\sim$)
Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model. Let $\mathrm{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ be $\mathcal{E}(\mathbf{Q}^\alpha)$, i.e., the interpretation of Leibniz equality at type α. We define $\mathsf{a} \dot\sim \mathsf{b}$ in $\mathcal{D}_\alpha$ iff $\upsilon(\mathrm{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$.

Before checking $\dot\sim$ is a congruence, we first show that it is at least reflexive.

Lemma 1.3.60 *Let $\mathcal{M}$ be a Σ-model. For each type α and $\mathsf{a} \in \mathcal{D}_\alpha$, we have $\mathsf{a} \dot\sim \mathsf{a}$.*

Proof: We need to check $\upsilon(\mathcal{E}(\mathbf{Q}^\alpha)@\mathsf{a}@\mathsf{a}) \equiv \mathtt{T}$. Let X_α be a variable of type α and φ be some assignment such that $\varphi(X) \equiv \mathsf{a}$. Let $\mathsf{r} := \mathcal{E}_\varphi(\lambda P_{\alpha\to o}.\neg(PX) \vee PX))$. For any $\mathsf{p} \in \mathcal{D}_{\alpha\to o}$, since $\mathcal{E}$ is an evaluation function, we have $\upsilon(\mathsf{r}@\mathsf{p}) \equiv \upsilon(\mathcal{E}_{\varphi,[\mathsf{p}/P]}(\neg(PX) \vee PX))$. As $\mathcal{M}$ is a Σ-model, we have $\upsilon(\mathcal{E}_{\varphi,[\mathsf{p}/P]}(\neg(PX) \vee PX)) \equiv \mathtt{T}$ since either $\upsilon(\mathcal{E}_{\varphi,[\mathsf{p}/P]}(PX)) \equiv \mathtt{T}$ or $\upsilon(\mathcal{E}_{\varphi,[\mathsf{p}/P]}(\neg(PX))) \equiv \mathtt{T}$. So, again since $\mathcal{M}$ is a Σ-model, $\upsilon(\mathcal{E}(\Pi^{\alpha\to o})@\mathsf{r}) \equiv \mathtt{T}$. By the definitions of r and $\doteq^\alpha$, we have $\upsilon(\mathcal{E}_\varphi(X \doteq^\alpha X)) \equiv \mathtt{T}$. As $X \doteq^\alpha X$ is a β-reduct of $\mathbf{Q}^\alpha XX$, we have $\upsilon(\mathcal{E}_\varphi(\mathbf{Q}^\alpha XX)) \equiv \mathtt{T}$ as well. Using $\varphi(X) \equiv \mathsf{a}$, we see that $\upsilon(\mathcal{E}(\mathbf{Q}^\alpha)@\mathsf{a}@\mathsf{a}) \equiv \mathtt{T}$. □

In order to check that $\dot\sim$ is a congruence, it is useful to unwind the definitions to better characterize when $\mathsf{a} \dot\sim \mathsf{b}$ for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$.

Lemma 1.3.61 (Properties of $\dot\sim$)
Let $\mathcal{M}$ be a Σ-model. For each type α and $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$, the following are equivalent:

1. $\mathsf{a} \dot\sim \mathsf{b}$.
2. *For all variables X_α and Y_α and assignments φ such that $\varphi(X) \equiv \mathsf{a}$ and $\varphi(Y) \equiv \mathsf{b}$, we have $\upsilon(\mathcal{E}_\varphi(X \doteq^\alpha Y)) \equiv \mathtt{T}$.*
3. *For every $\mathsf{p} \in \mathcal{D}_{\alpha\to o}$, $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \mathtt{T}$ implies $\upsilon(\mathsf{p}@\mathsf{b}) \equiv \mathtt{T}$.*
4. *For every $\mathsf{p} \in \mathcal{D}_{\alpha\to o}$, $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \upsilon(\mathsf{p}@\mathsf{b})$.*

Proof: At each type α, let $\mathrm{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ be the interpretation $\mathcal{E}(\mathbf{Q}^\alpha)$ of Leibniz equality. By definition, $\mathsf{a} \dot\sim \mathsf{b}$ iff $\upsilon(\mathrm{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$.

(1) implies (2): Suppose $\mathsf{a} \mathrel{\dot\sim} \mathsf{b}$ and φ is an assignment with $\varphi(X_\alpha) \equiv \mathsf{a}$ and $\varphi(Y_\alpha) \equiv \mathsf{b}$. Since $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$, we have $\upsilon(\mathcal{E}_\varphi(\mathbf{Q}^\alpha XY)) \equiv \mathtt{T}$. Since $\mathcal{E}$ respects β-equality (cf. 1.3.18), we have $\upsilon(\mathcal{E}_\varphi(X \doteq^\alpha Y)) \equiv \mathtt{T}$.

(2) implies (3): Suppose $\upsilon(\mathcal{E}_\varphi(X \doteq^\alpha Y)) \equiv \mathtt{T}$ whenever φ is an assignment with $\varphi(X) \equiv \mathsf{a}$ and $\varphi(Y) \equiv \mathsf{b}$. Let X and Y be particular distinct variables of type α and φ be any such assignment with $\varphi(X) \equiv \mathsf{a}$ and $\varphi(Y) \equiv \mathsf{b}$. Let $\mathsf{p} \in \mathcal{D}_{\alpha\to o}$ with $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \mathtt{T}$ and a variable $P_{\alpha\to o}$ be given. By assumption, $\upsilon(\mathcal{E}_\varphi(\forall P_{\alpha\to o}.\neg(PX) \lor (PY))) \equiv \mathtt{T}$. Since $\upsilon(\mathcal{E}_{\varphi,[\mathsf{p}/P]}(PX)) \equiv \upsilon(\mathsf{p}@\mathsf{a}) \equiv \mathtt{T}$, we have $\upsilon(\mathsf{p}@\mathsf{b}) \equiv \upsilon(\mathcal{E}_{\varphi,[\mathsf{p}/P]}(PY)) \equiv \mathtt{T}$.

(3) implies (4): Let $\mathsf{p} \in \mathcal{D}_{\alpha\to o}$ be given. If $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \mathtt{T}$, then we have $\upsilon(\mathsf{p}@\mathsf{b}) \equiv \mathtt{T}$ by assumption. So, $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \upsilon(\mathsf{p}@\mathsf{b})$ in this case. Otherwise, we must have $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \mathtt{F}$. Let $\mathsf{q} := \mathcal{E}_\varphi(\lambda X_\alpha.\neg(P_{\alpha\to o}X))$ where φ is some assignment with $\varphi(P) := \mathsf{p}$. Since $\mathcal{M}$ is a model, $\upsilon(\mathsf{q}@\mathsf{a}) \equiv \upsilon(\mathcal{E}(\neg)@(\mathsf{p}@\mathsf{a})) \equiv \mathtt{T}$. Applying the assumption to q, we have $\upsilon(\mathsf{q}@\mathsf{b}) \equiv \mathtt{T}$ and so $\upsilon(\mathcal{E}(\neg)@(\mathsf{p}@\mathsf{b})) \equiv \mathtt{T}$. Thus, $\upsilon(\mathsf{p}@\mathsf{b}) \equiv \mathtt{F}$ and $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \upsilon(\mathsf{p}@\mathsf{b})$ in this case as well.

(4) implies (1): Suppose $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \upsilon(\mathsf{p}@\mathsf{b})$ for every $\mathsf{p} \in \mathcal{D}_{\alpha\to o}$. In particular, this holds for $\mathsf{p} := \mathsf{q}^\alpha@\mathsf{a} \in \mathcal{D}_{\alpha\to o}$. Since $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{a}) \equiv \mathtt{T}$ by Lemma 1.3.60, we must have $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$. That is, $\mathsf{a} \mathrel{\dot\sim} \mathsf{b}$. □

Theorem 1.3.62 (Properties of $\mathcal{M}/_{\dot\sim}$) *Let $\mathcal{M}$ be a Σ-model. Then $\dot\sim$ is a congruence relation on the model $\mathcal{M}$ and $\mathcal{M}/_{\dot\sim}$ satisfies property $\mathfrak{q}$. Furthermore, if for every type α, $=^\alpha \in \Sigma_\alpha$ and $\upsilon(\mathcal{E}(=^\alpha)@a@b) \equiv \mathtt{T}$ iff $a \mathrel{\dot\sim} b$ for all $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$, then $\mathcal{M}/_{\dot\sim}$ is a Σ-model with primitive equality.*

Proof: We first verify that $\dot\sim$ is a congruence relation on each $\mathcal{D}_\alpha$. Reflexivity was shown in Lemma 1.3.60. To check symmetry and transitivity we use condition (4) in Lemma 1.3.61. For symmetry, let $\mathsf{a} \mathrel{\dot\sim} \mathsf{b}$ in $\mathcal{D}_\alpha$ and $\mathsf{p} \in \mathcal{D}_{\alpha\to o}$ be given. So, $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \upsilon(\mathsf{p}@\mathsf{b})$. Generalizing over p, we have $\mathsf{b} \mathrel{\dot\sim} \mathsf{a}$. For transitivity, let $\mathsf{a} \mathrel{\dot\sim} \mathsf{b}$ and $\mathsf{b} \mathrel{\dot\sim} \mathsf{c}$ in $\mathcal{D}_\alpha$ and $\mathsf{p} \in \mathcal{D}_{\alpha\to o}$ be given. So, $\upsilon(\mathsf{p}@\mathsf{a}) \equiv \upsilon(\mathsf{p}@\mathsf{b}) \equiv \upsilon(\mathsf{p}@\mathsf{c})$. Generalizing over p, we have $\mathsf{a} \mathrel{\dot\sim} \mathsf{c}$.

We next verify that $\dot\sim$ is a congruence. Suppose $\mathsf{f} \mathrel{\dot\sim} \mathsf{g}$ in $\mathcal{D}_{\alpha\to\beta}$ and $\mathsf{a} \mathrel{\dot\sim} \mathsf{b} \in \mathcal{D}_\alpha$. To show $\mathsf{f}@\mathsf{a} \mathrel{\dot\sim} \mathsf{g}@\mathsf{b}$ we use condition (3) in Lemma 1.3.61. Let $\mathsf{p} \in \mathcal{D}_{\beta\to o}$ with $\upsilon(\mathsf{p}@(\mathsf{f}@\mathsf{a})) \equiv \mathtt{T}$ be given. Let φ be an assignment with $\varphi(P_{\beta\to o}) \equiv \mathsf{p}$, $\varphi(X_\alpha) \equiv \mathsf{a}$ and $\varphi(G_{\alpha\to\beta}) \equiv \mathsf{g}$ for variables P, X and G. We can use Lemma 1.3.61(3) with $\mathcal{E}_\varphi(\lambda F_{\alpha\to\beta}.(P(FX)))$ and $\mathsf{f} \mathrel{\dot\sim} \mathsf{g}$ to verify that $\upsilon(\mathsf{p}@(\mathsf{g}@\mathsf{a})) \equiv \mathtt{T}$. Using Lemma 1.3.61(3) with $\mathcal{E}_\varphi(\lambda X_\alpha.(P(GX)))$ and $\mathsf{a} \mathrel{\dot\sim} \mathsf{b}$ verifies $\upsilon(\mathsf{p}@(\mathsf{g}@\mathsf{b})) \equiv \mathtt{T}$. So, $\mathsf{f}@\mathsf{a} \mathrel{\dot\sim} \mathsf{g}@\mathsf{b}$.

It remains to check that $\upsilon(\mathsf{a}) \equiv \upsilon(\mathsf{b})$ whenever $\mathsf{a} \mathrel{\dot\sim} \mathsf{b}$ for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_o$. Let $\mathsf{a} \mathrel{\dot\sim} \mathsf{b}$ in $\mathcal{D}_o$ be given. Applying Lemma 1.3.61(4) to $\mathcal{E}(\lambda X_o.X) \in \mathcal{D}_{o\to o}$ we have $\upsilon(\mathsf{a}) \equiv \upsilon(\mathcal{E}(\lambda X_o.X)@\mathsf{a}) \equiv \upsilon(\mathcal{E}(\lambda X_o.X)@\mathsf{b}) \equiv \upsilon(\mathsf{b})$ as desired. So, $\dot\sim$ is a congruence relation on $\mathcal{M}$.

Now, we show $\mathcal{M}/_{\dot\sim}$ satisfies property $\mathfrak{q}$. At each type α, let $\mathsf{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ be the interpretation $\mathcal{E}(\mathbf{Q}^\alpha)$ of Leibniz equality. To check property $\mathfrak{q}$, we show that $[\![\mathsf{q}^\alpha]\!]_{\dot\sim}$ is the appropriate object in $\mathcal{D}^{\dot\sim}_{\alpha\to\alpha\to o}$ for each $\alpha \in \mathcal{T}$. Let $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ be given. Note that $[\![\mathsf{a}]\!]_{\dot\sim} \equiv [\![\mathsf{b}]\!]_{\dot\sim}$ is equivalent to $\mathsf{a} \mathrel{\dot\sim} \mathsf{b}$. Also, $\upsilon^{\dot\sim}([\![\mathsf{q}^\alpha]\!]_{\dot\sim}@^{\dot\sim}[\![\mathsf{a}]\!]_{\dot\sim}@^{\dot\sim}[\![\mathsf{b}]\!]_{\dot\sim}) \equiv [\![\mathtt{T}]\!]_{\dot\sim}$ is equivalent to $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$. So, we need to show that $\upsilon(\mathsf{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$ iff $\mathsf{a} \mathrel{\dot\sim} \mathsf{b}$. But this is precisely the definition of $\dot\sim$.

The statement for primitive equality follows immediately by Theorem 1.3.55. □

Now, we know that when one takes a quotient of a model $\mathcal{M}$ by $\dot\sim$, one obtains a model satisfying property $\mathfrak{q}$. It is worthwhile to note the following relationship between $\dot\sim$ and property $\mathfrak{q}$.

Theorem 1.3.63 *Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model. The following are equivalent:*

1. *$\mathcal{M}$ satisfies property* q.
2. *For any congruence* $\sim$ *on* $\mathcal{M}$*, type* α*, and* $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$*,* $\mathsf{a} \sim \mathsf{b}$ *implies* $\mathsf{a} \equiv \mathsf{b}$.
3. *For any type* α*, and* $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$*,* $\mathsf{a} \dot{\sim} \mathsf{b}$ *implies* $\mathsf{a} \equiv \mathsf{b}$.
4. *For any type* α*,* $\mathfrak{L}^{\alpha}_{\equiv}(\mathcal{E}(\mathbf{Q}^{\alpha}))$ *holds for* υ.

Proof:

(1) implies (2) Suppose $\mathcal{M}$ satisfies q, $\sim$ is a congruence on $\mathcal{M}$, and $\mathsf{a} \sim \mathsf{b}$ for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$. Let $\mathrm{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ be the object at type α guaranteed to exist by property q. Since $\mathsf{a} \sim \mathsf{b}$, we have $(\mathrm{q}^\alpha @\mathsf{a}@\mathsf{a}) \sim (\mathrm{q}^\alpha @\mathsf{a}@\mathsf{b})$. By property q, we have $\upsilon(\mathrm{q}^\alpha @\mathsf{a}@\mathsf{a}) \equiv \mathtt{T}$ (since $\mathsf{a} \equiv \mathsf{a}$). Since $\sim$ is a congruence on the model, we have $\upsilon(\mathrm{q}^\alpha @\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$. By property q, this means $\mathsf{a} \equiv \mathsf{b}$.

(2) implies (3) This is obvious since $\dot{\sim}$ is a particular congruence on $\mathcal{M}$.

(3) implies (4) For each type α, we need to show $\mathfrak{L}^{\alpha}_{\equiv}(\mathcal{E}(\mathbf{Q}^{\alpha}))$ holds. By the definition of $\dot{\sim}$, for every $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ we have $\upsilon(\mathcal{E}(\mathbf{Q}^{\alpha})@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$, iff $\mathsf{a} \dot{\sim} \mathsf{b}$, iff $\mathsf{a} \equiv \mathsf{b}$. The last equivalence holds by our assumption that $\mathsf{a} \dot{\sim} \mathsf{b}$ implies that $\mathsf{a} \equiv \mathsf{b}$, and by Lemma 1.3.60.

(4) implies (1) For each type α, $\mathfrak{L}^{\alpha}_{\equiv}(\mathcal{E}(\mathbf{Q}^{\alpha}))$ implies $\mathcal{E}(\mathbf{Q}^{\alpha})$ is the witness required to show property q.

Remark 1.3.64 (Congruences for Σ-model with Primitive Equality)
Theorem 1.3.63 shows that once we have a model $\mathcal{M}$ which satisfies property q, there are no nontrivial congruences on $\mathcal{M}$. Hence, there are no nontrivial quotients of $\mathcal{M}$. In particular, the only possible congruence for a Σ-model with primitive equality is the trivial congruence given by the identity relation $\equiv$. Consequently, the quotient construction in the case of a Σ-model with primitive equality leads to essentially the same model again. We therefore do not consider quotients of models with primitive equality.

1.3.4 Σ-models over Frames

In this section, we define the notion of an isomorphism between two models and show every functional Σ-model is isomorphic to a model over a frame. In particular, this shows that the model class $\mathfrak{M}_{\beta\mathfrak{fb}}$ is simply the closure of the class $\mathfrak{H}$ of Henkin Models under isomorphism of Σ-models.

Definition 1.3.65 (Σ-Model Homomorphism)
Let $\mathcal{M}^1 \equiv (\mathcal{D}^1, @^1, \mathcal{E}^1, \upsilon^1)$ and $\mathcal{M}^2 \equiv (\mathcal{D}^2, @^2, \mathcal{E}^2, \upsilon^2)$ be Σ-models. A *homomorphism* from $\mathcal{M}^1$ to $\mathcal{M}^2$ is a typed function $\kappa: \mathcal{D}^1 \longrightarrow \mathcal{D}^2$ such that κ is a homomorphism from the evaluation $(\mathcal{D}^1, @^1, \mathcal{E}^1)$ to the evaluation $(\mathcal{D}^2, @^2, \mathcal{E}^2)$ and $\upsilon^1(\mathsf{a}) \equiv \upsilon^2(\kappa(\mathsf{a}))$ for every $\mathsf{a} \in \mathcal{D}^1_o$.

Definition 1.3.66 (Σ-Model Isomorphism)
Let $\mathcal{M}^1 \equiv (\mathcal{D}^1, @^1, \mathcal{E}^1, \upsilon^1)$ and $\mathcal{M}^2 \equiv (\mathcal{D}^2, @^2, \mathcal{E}^2, \upsilon^2)$ be Σ-models. An *isomorphism* from $\mathcal{M}^1$ to $\mathcal{M}^2$ is a homomorphism i from $\mathcal{M}^1$ to $\mathcal{M}^2$ such that there exists a homomorphism j from $\mathcal{M}^2$ to $\mathcal{M}^1$ where $j_\alpha: \mathcal{D}^2_\alpha \longrightarrow \mathcal{D}^1_\alpha$ is the inverse of $i_\alpha: \mathcal{D}^1_\alpha \longrightarrow \mathcal{D}^2_\alpha$ at each type α. Two models are said to be *isomorphic* if there is such an isomorphism. (It is clear from the definition that this is a symmetric relationship between models.)

Remark 1.3.67 The class $\mathfrak{H}$ of Henkin Models is not closed under isomorphism of models. Neither is the class $\mathfrak{ST}$ of standard models. This is because Henkin and standard models require that the domains $\mathcal{D}_{\alpha\to\beta}$ consist of functions from $\mathcal{F}(\mathcal{D}_\alpha;\mathcal{D}_\beta)$. We may, however, take a given Henkin model and appropriately modify it to obtain an isomorphic model that is not in the class of Henkin models. For example, we may choose $\mathcal{D}'_{\alpha\to\beta} := \{(0,f)|f \in \mathcal{D}_{\alpha\to\beta}\}$ and define @ appropriately (cf. Example 1.5.6 for a similar construction).

Lemma 1.3.68 *Let $\mathcal{M}^1$ and $\mathcal{M}^2$ be isomorphic Σ-models.*

1. *For any set of sentences Φ, $\mathcal{M}^1 \models \Phi$ implies $\mathcal{M}^2 \models \Phi$.*
2. *If $\mathcal{M}^1$ is a Σ-model with primitive equality, then $\mathcal{M}^2$ is a Σ-model with primitive equality.*
3. *If $* \in \{\mathsf{q}, \eta, \xi, \mathfrak{f}, \mathfrak{b}\}$ and $\mathcal{M}^1$ satisfies $*$, then $\mathcal{M}^2$ satisfies $*$.*

In particular, each model class $\mathfrak{M}_*$ is closed under isomorphism of models.

Proof: Let i be the homomorphism from $\mathcal{M}^1 \equiv (\mathcal{D}^1, @^1, \mathcal{E}^1, \upsilon^1)$ to $\mathcal{M}^2 \equiv (\mathcal{D}^2, @^2, \mathcal{E}^2, \upsilon^2)$ and j be its inverse.

1. Let Φ be a set of sentences with $\mathcal{M}^1 \models \Phi$. That is, for every $\mathbf{A} \in \Phi$, $\upsilon^1(\mathcal{E}^1(\mathbf{A})) \equiv \mathtt{T}$. So, for every $\mathbf{A} \in \Phi$, $\upsilon^2(\mathcal{E}^2(\mathbf{A})) \equiv \upsilon^1(j(\mathcal{E}^2(\mathbf{A}))) \equiv \upsilon^1(\mathcal{E}^1(\mathbf{A})) \equiv \mathtt{T}$ (since $\mathbf{A}$ is closed, we can ignore the variable assignment). This shows $\mathcal{M}^2 \models \Phi$.
2. Suppose $\mathcal{M}^1$ is a Σ-model with primitive equality. That is, $\mathfrak{L}^\alpha_=(\mathcal{E}^1(=^\alpha))$ holds for υ^1 at each type α. We need to show $\mathfrak{L}^\alpha_=(\mathcal{E}^2(=^\alpha))$ holds for υ^2 at each type α. Let $\mathsf{a}, \mathsf{b} \in \mathcal{D}^2_\alpha$ be given. Now, $\mathsf{a} \equiv \mathsf{b}$, iff $j(\mathsf{a}) \equiv j(\mathsf{b})$, iff $\upsilon^1(\mathcal{E}^1(=^\alpha)@^1 j(\mathsf{a})@^1 j(\mathsf{b})) \equiv \mathtt{T}$, iff $\upsilon^2(i(\mathcal{E}^1(=^\alpha)@^1 j(\mathsf{a})@^1 j(\mathsf{b}))) \equiv \mathtt{T}$, iff $\upsilon^2(\mathcal{E}^2(=^\alpha)@^2\mathsf{a}@^2\mathsf{b})) \equiv \mathtt{T}$.
3. We show the assertion separately for all the properties.

   q: Suppose $\mathcal{M}^1$ satisfies property q. Let α be a type and q^α be the witness for property q in $\mathcal{M}^1$ at α. We show that $i(\mathrm{q}^\alpha)$ is a witness for property q in $\mathcal{M}^2$ at α. Given $\mathsf{a}, \mathsf{b} \in \mathcal{D}^2_\alpha$. We have $\mathsf{a} \equiv \mathsf{b}$, iff $j(\mathsf{a}) \equiv j(\mathsf{b})$, iff $\upsilon^1(\mathrm{q}^\alpha @^1 j(\mathsf{a})@^1 j(\mathsf{b})) \equiv \mathtt{T}$, iff $\upsilon^2(i(\mathrm{q}^\alpha @^1 j(\mathsf{a})@^1 j(\mathsf{b}))) \equiv \mathtt{T}$, iff $\upsilon^2(i(\mathrm{q}^\alpha)@^2\mathsf{a}@^2\mathsf{b})) \equiv \mathtt{T}$. So, $\mathcal{M}^2$ satisfies property q.

   η: Suppose $\mathcal{M}^1$ satisfies property η. To show $\mathcal{M}^2$ satisfies η, let $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$ and an assignment φ into $\mathcal{M}^2$ be given. We compute

$$\begin{aligned}\mathcal{E}^2_\varphi(\mathbf{A}) &\equiv i \circ j(\mathcal{E}^2_\varphi(\mathbf{A}))\\ &\equiv i(\mathcal{E}^1_{j\circ\varphi}(\mathbf{A}))\\ &\equiv i(\mathcal{E}^1_{j\circ\varphi}(\mathbf{A}{\downarrow}_{\beta\eta}))\\ &\equiv i \circ j(\mathcal{E}^2_\varphi(\mathbf{A}{\downarrow}_{\beta\eta}))\\ &\equiv \mathcal{E}^2_\varphi(\mathbf{A}{\downarrow}_{\beta\eta})\end{aligned}$$

   So, $\mathcal{M}^2$ satisfies property η.

   ξ Suppose $\mathcal{M}^1$ satisfies property ξ. To show $\mathcal{M}^2$ satisfies ξ, let $\mathbf{M}, \mathbf{N} \in \mathit{wff}_\beta(\Sigma)$, a variable X_α, and an assignment ψ into $\mathcal{M}^2$ be given. Suppose $\mathcal{E}^2_{\psi,[\mathsf{b}/X]}(\mathbf{M}) \equiv \mathcal{E}^2_{\psi,[\mathsf{b}/X]}(\mathbf{N})$ for all $\mathsf{b} \in \mathcal{D}^2_\alpha$. For any $\mathsf{a} \in \mathcal{D}^1_\alpha$, we compute

$$\begin{aligned}
\mathcal{E}^1_{j\circ\psi,[\mathsf{a}/X]}(\mathbf{M}) &\equiv j(\mathcal{E}^2_{i\circ j\psi,[i(\mathsf{a})/X]}(\mathbf{M})) \\
&\equiv j(\mathcal{E}^2_{\psi,[i(\mathsf{a})/X]}(\mathbf{M})) \\
&\equiv j(\mathcal{E}^2_{\psi,[i(\mathsf{a})/X]}(\mathbf{N})) \\
&\equiv \mathcal{E}^1_{j\circ\psi,[\mathsf{a}/X]}(\mathbf{N})
\end{aligned}$$

Since $\mathcal{M}^1$ satisfies property ξ, we know $\mathcal{E}^1_{j\circ\psi}(\lambda X.\mathbf{M}) \equiv \mathcal{E}^1_{j\circ\psi}(\lambda X.\mathbf{N})$. Finally, we compute

$$\mathcal{E}^2_\psi(\lambda X.\mathbf{M}) \equiv i(\mathcal{E}^1_{j\circ\psi}(\lambda X.\mathbf{M})) \equiv i(\mathcal{E}^1_{j\circ\psi}(\lambda X.\mathbf{N})) \equiv \mathcal{E}^2_\psi(\lambda X.\mathbf{N})$$

So, $\mathcal{M}^2$ satisfies property ξ.

$\mathfrak{f}$: Suppose $\mathcal{M}^1$ satisfies property $\mathfrak{f}$ and we are given $\mathsf{f}, \mathsf{g} \in \mathcal{D}^2_{\alpha\to\beta}$ for types α and β. Suppose further that $\mathsf{f}@^2\mathsf{b} \equiv \mathsf{g}@^2\mathsf{b}$ for every $\mathsf{b} \in \mathcal{D}^2_\alpha$. It is enough to show $j(\mathsf{f}) \equiv j(\mathsf{g})$. This follows from property $\mathfrak{f}$ in $\mathcal{M}^1$ if we can show $j(\mathsf{f})@^1\mathsf{a} \equiv j(\mathsf{g})@^1\mathsf{a}$ for every $\mathsf{a} \in \mathcal{D}^1_\alpha$. So, let $\mathsf{a} \in \mathcal{D}^1_\alpha$ be given. We finish the proof by computing

$$\begin{aligned}
j(\mathsf{f})@^1\mathsf{a} &\equiv j(\mathsf{f})@^1 j \circ i(\mathsf{a}) \\
&\equiv j(\mathsf{f}@^2 i(\mathsf{a})) \\
&\equiv j(\mathsf{g}@^2 i(\mathsf{a})) \\
&\equiv j(\mathsf{g})@^1 j \circ i(\mathsf{a}) \\
&\equiv j(\mathsf{g})@^1\mathsf{a}
\end{aligned}$$

$\mathfrak{b}$: Finally, if $\mathcal{M}^1$ satisfies property $\mathfrak{b}$, then $\mathcal{D}^1_o$ has two elements. Since $i_o : \mathcal{D}^1_o \longrightarrow \mathcal{D}^2_o$ has inverse j_o, $\mathcal{D}^2_o$ must also have two elements. Thus, $\mathcal{M}^2$ satisfies property $\mathfrak{b}$. $\square$

Theorem 1.3.69 (Models Over Frames) *Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model which satisfies $\mathfrak{f}$ (i.e., $\mathcal{M}$ is functional). Then there is an isomorphic model $\mathcal{M}^{\mathit{fr}}$ over a frame.*

Proof: We define the model $\mathcal{M}^{\mathit{fr}} := (\mathcal{D}^{\mathit{fr}}, @^{\mathit{fr}}, \mathcal{E}^{\mathit{fr}}, \upsilon^{\mathit{fr}})$ by defining its components.

We first define the domains $\mathcal{D}^{\mathit{fr}}$ for $\mathcal{M}^{\mathit{fr}}$ by induction on types. We simultaneously define functions $i_\alpha : \mathcal{D}_\alpha \longrightarrow \mathcal{D}^{\mathit{fr}}_\alpha$ and $j_\alpha : \mathcal{D}^{\mathit{fr}}_\alpha \longrightarrow \mathcal{D}_\alpha$ which will witness that the two models are isomorphic. At each step of the definition, we check that i_α and j_α are mutual inverses. For base types $\alpha \in \{\iota, o\}$ let $\mathcal{D}^{\mathit{fr}}_\alpha := \mathcal{D}_\alpha$ and i_α and j_α be the identity functions (clearly mutual inverses).

Given two types α and β, we assume we have $\mathcal{D}^{\mathit{fr}}_\alpha$, mutual inverses $i_\alpha : \mathcal{D}_\alpha \to \mathcal{D}^{\mathit{fr}}_\alpha$ and $j_\alpha : \mathcal{D}^{\mathit{fr}}_\alpha \longrightarrow \mathcal{D}_\alpha$, as well as $\mathcal{D}^{\mathit{fr}}_\beta$ and mutual inverses $i_\beta : \mathcal{D}_\beta \to \mathcal{D}^{\mathit{fr}}_\beta$ and $j_\beta : \mathcal{D}^{\mathit{fr}}_\beta \longrightarrow \mathcal{D}_\beta$. We define

$$\mathcal{D}^{\mathit{fr}}_{\alpha\to\beta} := \left\{ f : \mathcal{D}^{\mathit{fr}}_\alpha \longrightarrow \mathcal{D}^{\mathit{fr}}_\beta \;\middle|\; \exists \mathsf{f} \in \mathcal{D}_{\alpha\to\beta}.\forall a \in \mathcal{D}^{\mathit{fr}}_\alpha . f(a) \equiv i_\beta(\mathsf{f}@j_\alpha(a)) \right\}$$

Note that $\mathcal{D}^{\mathit{fr}}_{\alpha\to\beta} \subseteq \mathcal{F}(\mathcal{D}^{\mathit{fr}}_\alpha; \mathcal{D}^{\mathit{fr}}_\beta)$. To define the map $i_{\alpha\to\beta} : \mathcal{D}_{\alpha\to\beta} \longrightarrow \mathcal{D}^{\mathit{fr}}_{\alpha\to\beta}$, we let $i_{\alpha\to\beta}(\mathsf{f})$ be the function taking each $a \in \mathcal{D}^{\mathit{fr}}_\alpha$ to $i_\beta(\mathsf{f}@j_\alpha(a))$. This choice for $i_{\alpha\to\beta}(\mathsf{f})$ is clearly in $\mathcal{D}^{\mathit{fr}}_{\alpha\to\beta}$ by definition. To define the inverse map $j_{\alpha\to\beta} : \mathcal{D}^{\mathit{fr}}_{\alpha\to\beta} \longrightarrow \mathcal{D}_{\alpha\to\beta}$, we must use the fact that $\mathcal{M}$ is functional. Given any $f \in \mathcal{D}^{\mathit{fr}}_{\alpha\to\beta}$, by definition there is some $\mathsf{f} \in \mathcal{D}_{\alpha\to\beta}$ such that $f(a) \equiv i_\beta(\mathsf{f}@j_\alpha(a))$ for every $a \in \mathcal{D}^{\mathit{fr}}_\alpha$. By functionality and the fact that the i and j at types α and β are already inverses, this f is unique, since if $i_\beta(\mathsf{f}@j_\alpha(a)) \equiv i_\beta(\mathsf{g}@j_\alpha(a))$ for every $a \in \mathcal{D}^{\mathit{fr}}_\alpha$, then

$\mathsf{f}@j_\alpha(i_\alpha(\mathsf{a})) \equiv \mathsf{g}@j_\alpha(i_\alpha(\mathsf{a}))$ for every $\mathsf{a} \in \mathcal{D}^{fr}_\alpha$. That is, $\mathsf{f}@\mathsf{a} \equiv \mathsf{g}@\mathsf{a}$ for every $\mathsf{a} \in \mathcal{D}^{fr}_\alpha$. So, for every $f \in \mathcal{D}^{fr}_{\alpha\to\beta}$, we define $j_{\alpha\to\beta}(f)$ to be the *unique* f such that $f(a) \equiv i_\beta(\mathsf{f}@j_\alpha(a))$. It is easy to check that $i_{\alpha\to\beta}$ and $j_{\alpha\to\beta}$ are mutually inverse.

For the applicative structure $(\mathcal{D}^{fr}, @^{fr})$ to be a frame, we are forced to let the application operator $@^{fr}$ to be function application. That is, for every $f \in \mathcal{D}^{fr}_{\alpha\to\beta}$ and $a \in \mathcal{D}^{fr}_\alpha$, $f@^{fr}a := f(a)$. We define the evaluation function $\mathcal{E}^{fr}$ simply by $\mathcal{E}^{fr}_\varphi(\mathbf{A}) := i(\mathcal{E}_{j\circ\varphi}(\mathbf{A}))$ for every $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$ and assignment φ into the applicative structure $(\mathcal{D}^{fr}, @^{fr})$. Since $\mathcal{D}^{fr}_o \equiv \mathcal{D}_o$, we can let $\upsilon^{fr} := \upsilon$.

We only sketch the remainder of the proof. First one can show that i and j preserve application. One can use this fact to verify that $\mathcal{E}^{fr}$ is an evaluation function so that $(\mathcal{D}^{fr}, @^{fr}, \mathcal{E}^{fr})$ is a Σ-evaluation, and that $\upsilon^{fr} \equiv \upsilon$ is a valuation function for this evaluation. This verifies $\mathcal{M}^{fr}$ is a model. Finally, to verify one has an isomorphism, one can easily check the remainder of the conditions for i and j to be homomorphisms between the models. These are isomorphisms since they are mutually inverse on the domains of each type.

$\square$

We can conclude that $\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$ is simply the closure of the class of $\mathfrak{H}$ of Henkin models under isomorphism. Given any $\mathcal{M} \in \mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$, by Theorem 1.3.69, there is an isomorphic model $\mathcal{M}^{fr}$ over a frame. By Lemma 1.3.68, this model $\mathcal{M}^{fr}$ satisfies $\mathfrak{q}$, $\mathfrak{f}$, and $\mathfrak{b}$ (since $\mathcal{M}$ does). Also, if primitive equality is present in the signature, by the same lemma we know $\mathcal{M}^{fr}$ is a model with primitive equality. That is, $\mathcal{M}^{fr} \in \mathfrak{H}$.

1.4 Properties of Model Classes

In this section we discuss some properties of the model classes introduced in section 1.3. Our interest is in the properties of Leibniz equality and primitive equality.

Definition 1.4.1 (Extensionality for Leibniz Equality)
We call a formula of the form

$$\mathrm{EXT}^{\alpha\to\beta}_{\doteq} \quad := \quad \forall F_{\alpha\to\beta}.\forall G_{\alpha\to\beta}.(\forall X_\alpha.FX \doteq^\beta GX) \Rightarrow F \doteq^{\alpha\to\beta} G$$

an *axiom of (strong) functional extensionality for Leibniz equality*, and refer to the set

$$\mathrm{EXT}^{\to}_{\doteq} := \{\mathrm{EXT}^{\alpha\to\beta}_{\doteq} \mid \alpha, \beta \in \mathcal{T}\}$$

as the *axioms of (strong) functional extensionality for Leibniz equality.* Note that $\mathrm{EXT}^{\to}_{\doteq}$ specifies functionality of the relation corresponding to Leibniz equality $\doteq$. We call the formula

$$\mathrm{EXT}^{\to}_{\doteq} \quad := \quad \{\mathrm{EXT}^{\alpha\to\beta}_{\doteq} \mid \alpha, \beta \in \mathcal{T}\}$$

as the *axioms of (strong) functional extensionality for Leibniz equality.* Note that $\mathrm{EXT}^{\to}_{\doteq}$ specifies functionality of the relation corresponding to Leibniz equality $\doteq$. We call the formula

$$\mathrm{EXT}^{o}_{\doteq} \quad := \quad \forall A_o.\forall B_o.(A \Leftrightarrow B) \Rightarrow A \doteq^o B$$

the *axiom of Boolean extensionality.* We call the set $\mathrm{EXT}^{\to}_{\doteq} \cup \{\mathrm{EXT}^{o}_{\doteq}\}$ the *axioms of (strong) extensionality for Leibniz equality.*

In examples 1.5.4 to 1.5.7 below we give concrete models in which $\mathrm{EXT}^{o}_{\doteq}$ and $\mathrm{EXT}^{\alpha\to\beta}_{\doteq}$ fail in various ways. First, we prove relationships between properties $\mathfrak{q}$, $\mathfrak{b}$ and $\mathfrak{f}$ and the statements $\mathrm{EXT}^{o}_{\doteq}$ and $\mathrm{EXT}^{\to}_{\doteq}$.

Lemma 1.4.2 (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 $\mathbf{A}, \mathbf{B} \in wff_\alpha(\Sigma)$.*

1. *If $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{B})$, then $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) \equiv \mathtt{T}$.*
2. *If $\mathcal{M}$ satisfies property $\mathfrak{q}$ and $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) \equiv \mathtt{T}$, then $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{B})$.*

Proof: Let φ be any assignment into $\mathcal{M}$.

1. Suppose $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{B})$. Given $\mathsf{r} \in \mathcal{D}_{\alpha\to o}$, we have either $\upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{A})) \equiv \upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{F}$ or $\upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{B})) \equiv \upsilon(\mathsf{r}@\mathcal{E}_\varphi(\mathbf{A})) \equiv \mathtt{T}$. In either case, for any variable $P_{\alpha\to o}$ not in $free(\mathbf{A}) \cup free(\mathbf{B})$, we have $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/P]}(\neg(P\mathbf{A}) \vee P\mathbf{B})) \equiv \mathtt{T}$. So, we have $\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B}) \equiv \mathtt{T}$.
2. Suppose $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) \equiv \mathtt{T}$. By property $\mathfrak{q}$, there is some $\mathrm{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that for $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ we have $\upsilon(\mathrm{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$ iff $\mathsf{a} \equiv \mathsf{b}$. Let $\mathsf{r} \equiv \mathrm{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})$. From $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) \equiv \mathtt{T}$, we obtain $\mathcal{E}_{\varphi,[\mathsf{r}/P]}(\neg P\mathbf{A} \vee P\mathbf{B}) \equiv \mathtt{T}$ (where $P_{\alpha\to o} \notin free(\mathbf{A}) \cup free(\mathbf{B})$). Since $\mathcal{E}_{\varphi,[\mathsf{r}/P]}(P\mathbf{A}) \equiv \mathrm{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})@\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathtt{T}$, we must have $\upsilon(\mathcal{E}_{\varphi,[\mathsf{r}/P]}(P\mathbf{B})) \equiv \mathtt{T}$. That is, $\upsilon(\mathrm{q}^\alpha@\mathcal{E}_\varphi(\mathbf{A})@\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{T}$. By the choice of q^α, we have $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{B})$. □

in	$\mathfrak{M}_\beta$, $\mathfrak{M}_{\beta\eta}$, $\mathfrak{M}_{\beta\xi}$		$\mathfrak{M}_{\beta\mathfrak{f}}$		$\mathfrak{M}_{\beta\mathfrak{b}}$, $\mathfrak{M}_{\beta\eta\mathfrak{b}}$, $\mathfrak{M}_{\beta\xi\mathfrak{b}}$		$\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$	
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.

Figure 1.5: Extensionality in Σ-models

Theorem 1.4.3 (Extensionality in Σ-models) *Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model.*

1. *If $\mathcal{M}$ satisfies property $\mathfrak{q}$ but not property $\mathfrak{f}$, then $\mathcal{M} \not\models \mathrm{EXT}^{\to}_{\doteq}$.*
2. *If $\mathcal{M}$ satisfies property $\mathfrak{q}$ but not property $\mathfrak{b}$, then $\mathcal{M} \not\models \mathrm{EXT}^{o}_{\doteq}$.*
3. *If $\mathcal{M}$ satisfies properties $\mathfrak{q}$ and $\mathfrak{f}$, then $\mathcal{M} \models \mathrm{EXT}^{\to}_{\doteq}$.*
4. *If $\mathcal{M}$ satisfies property $\mathfrak{b}$, then $\mathcal{M} \models \mathrm{EXT}^{o}_{\doteq}$.*

Thus we can characterize the different semantical structures with respect to Boolean and functional extensionality in the table in Figure 1.5.

Proof:

1. Suppose $\mathcal{M}$ satisfies property $\mathfrak{q}$ but does not satisfy property $\mathfrak{f}$. Then there must be types α and β and objects $\mathsf{f}, \mathsf{g} \in \mathcal{D}_{\alpha\to\beta}$ such that $\mathsf{f} \not\equiv \mathsf{g}$ but $\mathsf{f}@\mathsf{a} \equiv \mathsf{g}@\mathsf{a}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$. 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 $\varphi(F) \equiv \mathsf{f}$ and $\varphi(G) \equiv \mathsf{g}$. For any $\mathsf{a} \in \mathcal{D}_\alpha$, $\mathsf{f}@\mathsf{a} \equiv \mathsf{g}@\mathsf{a}$ implies $\upsilon(\mathcal{E}_{\varphi,[\mathsf{a}/X]}(FX \doteq^\beta GX)) \equiv \mathtt{T}$ by Lemma 1.4.2(1). Using the fact that υ is a valuation, we have $\upsilon(\mathcal{E}_\varphi(\forall X.(FX \doteq^\beta GX))) \equiv \mathtt{T}$. On the other hand, since $\mathsf{f} \not\equiv \mathsf{g}$ and $\mathcal{M}$ satisfies property $\mathfrak{q}$, we have $\upsilon(\mathcal{E}_\varphi(F \doteq^{\alpha\to\beta} G)) \equiv \mathtt{F}$ by contraposition of Lemma 1.4.2(2). This implies $\mathcal{M} \not\models \mathrm{EXT}^{\alpha\to\beta}_{\doteq}$.

2. 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 $\mathsf{a}, \mathsf{b} \in \mathcal{D}_o$ such that $\upsilon(\mathsf{a}) \equiv \upsilon(\mathsf{b})$.

   Let $A_o, B_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$ with $\varphi(A) \equiv \mathsf{a}$ and $\varphi(B) \equiv \mathsf{b}$. By Lemma 1.3.48, we know $\upsilon(\mathcal{E}_\varphi(A \Leftrightarrow B)) \equiv \mathtt{T}$. Since $\mathsf{a} \not\equiv \mathsf{b}$ and property $\mathfrak{q}$ holds, by contraposition of Lemma 1.4.2(2), we know $\upsilon(\mathcal{E}_\varphi(A \doteq^o B)) \equiv \mathtt{F}$. It follows that $\mathcal{M} \not\models \mathrm{EXT}^o_{\doteq}$.

3. Let φ be any assignment into $\mathcal{M}$. From $\upsilon(\mathcal{E}_\varphi(\forall X_\alpha. FX \doteq GX)) \equiv \mathtt{T}$ we know $\upsilon(\mathcal{E}_{\varphi,[\mathsf{a}/X]}(FX \doteq GX)) \equiv \mathtt{T}$ holds for all $\mathsf{a} \in \mathcal{D}_\alpha$. By Lemma 1.4.2(2) we can conclude that $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(FX) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(GX)$ for all $\mathsf{a} \in \mathcal{D}_\alpha$ and hence $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(F)@\mathcal{E}_{\varphi,[\mathsf{a}/X]}(X) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(G)@\mathcal{E}_{\varphi,[\mathsf{a}/X]}(X)$ for all $\mathsf{a} \in \mathcal{D}_\alpha$. That is, $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(F)@\mathsf{a} \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(G)@\mathsf{a}$ for all $\mathsf{a} \in \mathcal{D}_\alpha$. Since X does not occur free in F or G, by property $\mathfrak{f}$ and Definition 1.3.17(3) we obtain $\mathcal{E}_\varphi(F) \equiv \mathcal{E}_\varphi(G)$. This finally gives us that $\upsilon(\mathcal{E}_\varphi(F \doteq^{\alpha\to\beta} G)) \equiv \mathtt{T}$ with Lemma 1.4.2(1). It follows that $\mathcal{M} \models \mathrm{EXT}^{\alpha\to\beta}_{\doteq}$ and $\mathcal{M} \models \mathrm{EXT}^{\to}_{\doteq}$, since α and β were chosen arbitrarily.

   Note that we certainly need the assumption that $\mathcal{M}$ satisfies property $\mathfrak{q}$ (which is employed within the application of Lemma 1.4.2(2). As explained in Remark 1.3.52, there is a functional model in which property $\mathfrak{q}$ fails and $\mathrm{EXT}^{\iota\to\iota}_{\doteq}$ is not valid.

4. Let $A_o, B_o \in \mathcal{V}_o$ be distinct variables and φ be any assignment into $\mathcal{M}$. Since $\mathfrak{b}$ holds, we can assume $\mathcal{D}_o \equiv \{\mathtt{T}, \mathtt{F}\}$ and υ is the identity function. Suppose $\upsilon(\mathcal{E}_\varphi(A \Leftrightarrow B)) \equiv \mathtt{T}$. By Lemma 1.3.48, we have $\mathcal{E}_\varphi(A) \equiv \upsilon(\mathcal{E}_\varphi(A)) \equiv \upsilon(\mathcal{E}_\varphi(B)) \equiv \mathcal{E}_\varphi(B)$. By Lemma 1.4.2(1), we have $\upsilon(\mathcal{E}_\varphi(A \doteq^o B)) \equiv \mathtt{T}$. It follows that $\mathcal{M} \models \mathrm{EXT}^o_{\doteq}$. $\Box$

Remark 1.4.4 (Alternative Definitions of Equality) Leibniz equality is a very prominent way of defining equality in higher order logic. However, there are alternative definitions such as (cf. [And02, p. 203])

$$\ddot{=}^\alpha \;:=\; \lambda X_\alpha Y_\alpha. \forall Q_{\alpha\to\alpha\to o}. (\forall Z_\alpha. QZZ) \Rightarrow QXY$$

An important question is whether an alternative definition of equality is equivalent to the Leibniz definition in particular model classes. As Remark 1.3.47 shows, this has to be carefully investigated for each equality definition and each model class in question. We can show that for all $\mathbf{A}_\alpha, \mathbf{B}_\alpha \in cwff_\alpha(\Sigma)$ $\mathbf{A} \ddot{=} \mathbf{B}$ and $\mathbf{A} \doteq \mathbf{B}$ are equivalent modulo υ for all $\mathcal{M} \in \mathfrak{M}_\beta$ (and thus for all other model classes). That is, we can show $\upsilon(\mathcal{E}(\mathbf{A} \ddot{=}^\alpha \mathbf{B})) \equiv \upsilon(\mathcal{E}(\mathbf{A} \doteq^\alpha \mathbf{B}))$. Note that this is weaker than showing $\mathcal{E}(\mathbf{A} \ddot{=}^\alpha \mathbf{B}) \equiv \mathcal{E}(\mathbf{A} \doteq^\alpha \mathbf{B})$. The key idea is to reduce the definition of $\ddot{=}$ to $\doteq$ (and vice versa) by instantiating the universally quantified set variables Q and P appropriately. We may, for instance, show $\mathbf{A} \ddot{=}^\alpha \mathbf{B}$ implies $\mathbf{A} \doteq^\alpha \mathbf{B}$, by choosing the instantiation $[\lambda U_\alpha V_\alpha. \forall P_{\alpha\to o}. PU \Rightarrow PV/Q]$ and the converse with $[\lambda V_\alpha. \forall Q_{\alpha\to\alpha\to o}. (\forall Z_\alpha. QZZ) \Rightarrow Q\mathbf{A}V/P]$. As a consequence the properties of Leibniz equality with respect to extensionality also apply to $\ddot{=}$.

Definition 1.4.5 (Extensionality for Primitive Equality) Analogous to the extensionality axioms for Leibniz equality, we can define the *axioms of strong (functional and Boolean) extensionality for primitive equality*:

$$\begin{aligned} \mathrm{EXT}^{\alpha\to\beta}_{=} &:= \forall F_{\alpha\to\beta}. \forall G_{\alpha\to\beta}. (\forall X_\alpha. FX =^\beta GX) \Rightarrow F =^{\alpha\to\beta} G \\ \mathrm{EXT}^{o}_{=} &:= \forall A_o. \forall B_o. (A \Leftrightarrow B) \Rightarrow A =^o B \end{aligned}$$

As before we refer to the set $\mathrm{EXT}^{\to}_{=} := \{\mathrm{EXT}^{\alpha\to\beta}_{=} \mid \alpha, \beta \in \mathcal{T}\}$ as the *axioms of (strong) functional extensionality for primitive equality.*

The following lemma shows that in a Σ-model with primitive equality for each $\alpha \in \mathcal{T}$ the denotations of $=^\alpha$ and $\doteq^\alpha$ are identical modulo υ.

Lemma 1.4.6 (Primitive and Leibniz equality)
If $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_*$ is a Σ-model with primitive equality, then we have $\upsilon(\mathcal{E}_\varphi(\mathbf{A} =^\alpha \mathbf{B})) \equiv \upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B}))$ for all assignments φ into $\mathcal{M}$, types $\alpha \in \mathcal{T}$, and $\mathbf{A}, \mathbf{B} \in wff_\alpha(\Sigma)$.

Proof: Since property $\mathfrak{q}$ holds for $\mathcal{M} \in \mathfrak{M}_*$, by Lemma 1.4.2 parts (1) and (2), we have $\upsilon(\mathcal{E}_\varphi(\mathbf{A} \doteq^\alpha \mathbf{B})) \equiv \mathtt{T}$ iff $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{B})$. Since $\mathcal{M}$ is a Σ-model with primitive equality, we know $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{B})$ is equivalent to $\upsilon(\mathcal{E}(=^\alpha)@\mathcal{E}_\varphi(\mathbf{A})@\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{T}$, and hence to $\upsilon(\mathcal{E}_\varphi(\mathbf{A} =^\alpha \mathbf{B})) \equiv \mathtt{T}$. □

Remark 1.4.7 Lemma 1.4.6 implies that for all models in our model classes $\mathfrak{M}_*$ the extensionality axioms for primitive equality are equivalent to the corresponding extensionality axioms for Leibniz equality. Thus, the analysis for the Leibniz versions applies directly to the versions using primitive equality. Also, Lemma 1.4.6 reinforces that (provided property $\mathfrak{q}$ holds) we can indeed use Leibniz equality to treat equality as a defined notion (relative to models in $\mathfrak{M}_*$). Thus, we principally do not need to assume the constants $=^\alpha$ to be in our signature. The critical part in this choice is that for ensuring the correct meaning for $\mathbf{Q}^\alpha$ we have to require the existence of an object representing the identity relation for each type in each Σ-model (cf. [And72a] for a discussion in the context of Henkin models). This requirement is automatically met if we consider primitive equality. Hence it seems natural to treat equality as primitive.

Remark 1.4.8 (Properties η and ξ) We have shown, in the presence of property $\mathfrak{q}$, a model $\mathcal{M}$ satisfies property $\mathfrak{f}$ iff $\mathcal{M} \models \mathrm{EXT}^{\rightarrow}_{\doteq}$. Similarly, we have shown that property $\mathfrak{b}$ corresponds to a model satisfying $\mathrm{EXT}^{o}_{\doteq}$. A corresponding analysis can be done for properties η and ξ (cf. Definition 1.3.46). Assume $\mathcal{M}$ satisfies property $\mathfrak{q}$. Then, $\mathcal{M}$ satisfies property η iff $\mathcal{M} \models \mathbf{A} \doteq^\alpha (\mathbf{A}{\downarrow_{\beta\eta}})$ for every type α and closed formula $\mathbf{A} \in cwff_\alpha(\Sigma)$. Also, $\mathcal{M}$ satisfies property ξ iff

$$\mathcal{M} \models \forall F_{\alpha\to\beta}.\forall G_{\alpha\to\beta}.(\forall X_\alpha.FX \doteq^\beta GX) \Rightarrow (\lambda X.FX) \doteq^{\alpha\to\beta} (\lambda X.GX)$$

for all types α and β.

1.5 Example Models

We now sketch the construction of models in the model classes $\mathfrak{M}_*$ to demonstrate concretely how properties for Boolean, strong and weak functional extensionality can fail. We need this to show that the inclusions of the model classes defined in Section 1.3 are proper, and we indeed need all of them.

We start with the simplest example of a Henkin model, which we will call the *singleton model*, since the domain of individuals is a singleton. Note that the underlying evaluation of this model is not the singleton evaluation from Example 1.3.28 since $\mathcal{D}_o$ has two elements. In this model, all forms of extensionality are valid.

Example 1.5.1 (Singleton Model — $\mathcal{M}^{\beta\mathfrak{fb}} \in \mathfrak{ST} \subseteq \mathfrak{H} \subseteq \mathfrak{M}_{\beta\mathfrak{fb}}$)
Let $(\mathcal{D}, @)$ be the full frame with $\mathcal{D}_o := \{\mathtt{T}, \mathtt{F}\}$ and $\mathcal{D}_\iota := \{*\}$. One can easily define an evaluation function $\mathcal{E}$ for this frame by induction on terms, using functions to interpret λ-abstractions. The identity function $\upsilon{:}\,\mathcal{D}_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ is a valuation, assuming the logical constants are interpreted in the standard way (including primitive equality, if present in Σ). So, $\mathcal{M}^{\beta\mathfrak{fb}} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ defines

a model. This model clearly satisfies all our properties $\mathfrak{b}$, $\mathfrak{f}$ (hence η and ξ) and $\mathfrak{q}$ (since the frame is full). So, $\mathcal{M}^{\beta\mathfrak{fb}} \in \mathfrak{ST} \subseteq \mathfrak{H} \subseteq \mathfrak{M}_{\beta\mathfrak{fb}}$.

Remark 1.5.2 In particular, all our model classes are non-empty. We also know that $\mathfrak{ST} \not\equiv \mathfrak{H} \not\equiv \mathfrak{M}_{\beta\mathfrak{fb}}$, so the inclusions are proper. By parts (3) and (4) of Theorem 1.4.3, we have $\mathcal{M}^{\beta\mathfrak{fb}} \models \mathrm{EXT}^{o}_{\doteq}$ and $\mathcal{M}^{\beta\mathfrak{fb}} \models \mathrm{EXT}^{\rightarrow}_{\doteq}$.

We can use the singleton model $\mathcal{M}^{\beta\mathfrak{fb}}$ to construct another model which makes the importance of property $\mathfrak{q}$ clear.

Remark 1.5.3 Let $\mathcal{M}^{\beta\mathfrak{fb}} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ as above and $\mathcal{TE}(\Sigma)^{\beta} \equiv (\mathcal{D}^{\beta}, @^{\beta}, \mathcal{E}^{\beta})$ be the β-term evaluation as defined in Definition 1.3.35. Let $\upsilon' : \mathcal{D}^{\beta}_{o} \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ be the function $\upsilon'(\mathbf{A}) := \upsilon(\mathcal{E}(\mathbf{A}))$ for every $\mathbf{A} \in cwff_{o}(\Sigma)\!\downarrow_{\beta}$. One can show $\mathcal{M}' := (\mathcal{D}^{\beta}, @^{\beta}, \mathcal{E}^{\beta}, \upsilon')$ is a Σ-model such that $\mathcal{M}' \models \mathbf{A}$ iff $\mathcal{M}^{\beta\mathfrak{fb}} \models \mathbf{A}$ for every sentence $\mathbf{A}$. In particular, $\mathcal{M}' \models \mathrm{EXT}^{o}_{\doteq}$ and $\mathcal{M}' \models \mathrm{EXT}^{\rightarrow}_{\doteq}$.

Nevertheless, $\mathcal{M}'$ fails to satisfy properties $\mathfrak{q}$, $\mathfrak{b}$, η and $\mathfrak{f}$. Property $\mathfrak{b}$ does not hold since $\mathcal{D}^{\beta}_{o} \equiv cwff_{o}(\Sigma)\!\downarrow_{\beta}$ is infinite. Property η does not hold since, for example,

$$\mathcal{E}^{\beta}(\lambda F_{\iota\rightarrow\iota} X_{\iota} . F X) \equiv \lambda F_{\iota\rightarrow\iota} X_{\iota} . F X \not\equiv \lambda F_{\iota\rightarrow\iota} . F \equiv \mathcal{E}^{\beta}(\lambda F_{\iota\rightarrow\iota} . F)$$

Property $\mathfrak{f}$ cannot hold since property η does not hold. (On the other hand, property ξ does hold since the underlying evaluation is a term evaluation.)

We know now by Theorem 1.4.3, either part (1) or part (2), that property $\mathfrak{q}$ must not hold. A concrete way to see that property $\mathfrak{q}$ fails is to consider two distinct constants $a_{\iota}, b_{\iota} \in \Sigma_{\iota}$. We must have $\mathcal{M}^{\beta\mathfrak{fb}} \models a \doteq^{\iota} b$ (since $\mathcal{D}_{\iota}$ has only one element), and so $\mathcal{M}' \models a \doteq^{\iota} b$. On the other hand a and b are distinct elements (as distinct β-normal forms) in $\mathcal{D}^{\beta}_{\iota}$.

The model $\mathcal{M}'$ shows that property $\mathfrak{q}$ is needed in the proofs of parts (1) and (2) of Theorem 1.4.3.

Example 1.5.4 (Failure of $\mathfrak{b}$ — $\mathcal{M}^{\beta\mathfrak{f}} \in \mathfrak{M}_{\beta\mathfrak{f}} \setminus \mathfrak{M}_{\beta\mathfrak{fb}}$)
Let $(\mathcal{D}, @)$ be the full frame with $\mathcal{D}_{o} = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}$ and $\mathcal{D}_{\iota} = \{0, 1\}$. We define an evaluation function $\mathcal{E}$ for this frame by defining $\mathcal{E}(\neg)$, $\mathcal{E}(\vee)$, and $\mathcal{E}(\Pi^{\alpha})$ to be the functions given in the following table:

$\mathcal{E}(\neg)$	a	b	c
	c	c	a

$\mathcal{E}(\vee)$	a	b	c
a	a	a	a
b	a	a	a
c	a	a	c

$$\mathcal{E}(\Pi^{\alpha})@\mathsf{f} = \begin{cases} \mathsf{a}, & \text{if } \mathsf{f}@\mathsf{g} \in \{\mathsf{a}, \mathsf{b}\} \text{ for all } \mathsf{g} \in \mathcal{D}_{\alpha} \\ \mathsf{c}, & \text{if } \mathsf{f}@\mathsf{g} = \mathsf{c} \text{ for some } \mathsf{g} \in \mathcal{D}_{\alpha} \end{cases}$$

We can choose $\mathcal{E}(w)$ to be arbitrary for parameters $w \in \Sigma$. Since the applicative structure $(\mathcal{D}, @)$ is a frame, hence functional, this uniquely determines $\mathcal{E}$ on all formulae. Also, since the frame is full, we are guaranteed that there will be enough functions to interpret λ-abstractions.

Let the map $\upsilon : \mathcal{D}_{o} \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ be defined by $\upsilon(\mathsf{a}) := \mathtt{T}$, $\upsilon(\mathsf{b}) := \mathtt{T}$ and $\upsilon(\mathsf{c}) := \mathtt{F}$. It is easy to check that $\mathcal{M}^{\beta\mathfrak{f}} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$ is indeed a Σ-model. Since this is a model over a frame, we automatically know it satisfies property $\mathfrak{f}$. Since the frame is full, we know property $\mathfrak{q}$ holds. (By the same argument, if primitive equality is in the signature, we can ensure $\mathcal{E}(=^{\alpha})$ is interpreted appropriately for each type α.) Clearly property $\mathfrak{b}$ fails, so we have $\mathcal{M}^{\beta\mathfrak{f}} \in \mathfrak{M}_{\beta\mathfrak{f}} \setminus \mathfrak{M}_{\beta\mathfrak{fb}}$. By Theorem 1.4.3(2), $\mathcal{M}^{\beta\mathfrak{f}} \not\models \mathrm{EXT}^{o}_{\doteq}$.

In this model one can easily verify, if $\mathsf{d} := \mathcal{E}_\varphi(\mathbf{D}_o)$ and $\mathsf{e} := \mathcal{E}_\varphi(\mathbf{E}_o)$, then the values $\mathcal{E}_\varphi(\mathbf{D} \wedge \mathbf{E})$, $\mathcal{E}_\varphi(\mathbf{D} \Rightarrow \mathbf{E})$, and $\mathcal{E}_\varphi(\mathbf{D} \Leftrightarrow \mathbf{E})$ are given by the following tables:

	e:		
$\mathcal{E}(\mathbf{D} \wedge \mathbf{E})$	a	b	c
d: a	a	a	c
b	a	a	c
c	c	c	c

	e:		
$\mathcal{E}(\mathbf{D} \Rightarrow \mathbf{E})$	a	b	c
d: a	a	a	c
b	a	a	c
c	a	a	a

	e:		
$\mathcal{E}(\mathbf{D} \Leftrightarrow \mathbf{E})$	a	b	c
d: a	a	a	c
b	a	a	c
c	c	c	a

Note that one can properly model the *woodchuck/groundhog* example from [LP00] referred to in the introduction in $\mathcal{M}^{\beta\mathfrak{f}}$.

Example 1.5.5 (Groundhogs and Woodchucks)
Let $\mathcal{M}^{\beta\mathfrak{f}}$ be given as above and suppose $woodchuck_{\iota\to o}$, $groundhog_{\iota\to o}$, $john_\iota$, and $phil_\iota$ are in the signature Σ. Let $\mathcal{E}(phil) := 0$ and $\mathcal{E}(john) := 1$. Let $\mathcal{E}(woodchuck)$ be the function $\mathsf{w} \in \mathcal{D}_{\iota\to o}$ with $\mathsf{w}(0) \equiv \mathsf{b}$ and $\mathsf{w}(1) \equiv \mathsf{c}$. Let $\mathcal{E}(groundhog)$ be the function $\mathsf{g} \in \mathcal{D}_{\iota\to o}$ with $\mathsf{g}(0) \equiv \mathsf{a}$ and $\mathsf{g}(1) \equiv \mathsf{c}$. One can show that the sentence $\forall X_\iota\text{.}(woodchuck\ X) \Leftrightarrow (groundhog\ X)$ is valid. Also, $\mathcal{E}(woodchuck\ phil) \equiv \mathsf{b}$ and $\mathcal{E}(groundhog\ phil) \equiv \mathsf{a}$, so the propositions $(woodchuck\ phil)$ and $(groundhog\ phil)$ are valid. Next, suppose $believe_{\iota\to o\to o} \in \Sigma$ and $\mathcal{E}(believe)$ is the (Curried) function $\mathsf{bel} \in \mathcal{D}_{\iota\to o\to o}$ such that $\mathsf{bel}(1)(\mathsf{b}) \equiv \mathsf{b}$ and $\mathsf{bel}(1)(\mathsf{a}) \equiv \mathsf{bel}(1)(\mathsf{c}) \equiv \mathsf{bel}(0)(\mathsf{a}) \equiv \mathsf{bel}(0)(\mathsf{b}) \equiv \mathsf{bel}(0)(\mathsf{c}) \equiv \mathsf{c}$ (Intuitively, John believes propositions with value b, but not those with value a or c). So, *believes john*$(woodchuck\ phil)$ is valid, while *believes john*$(groundhog\ phil)$ is not.

As we have seen, Boolean extensionality fails when one has more than two values in $\mathcal{D}_o$. We can generalize the construction defining $\mathcal{D}_o := \{\mathtt{F}\} \cup \mathcal{B}$, where $\mathcal{B}$ is any set with $\mathtt{T} \in \mathcal{B}$ and $\mathtt{F} \notin \mathcal{B}$. The model will satisfy Boolean extensionality iff $\mathcal{B} \equiv \{\mathtt{T}\}$. In this way, we can easily construct models for the case with property $\mathfrak{b}$ and the case without property $\mathfrak{b}$ simultaneously. We will use this idea to parameterize the remaining model constructions by $\mathcal{B}$. These semantic construction are similar to those in multi-valued logics, which have been studied for higher-order logic in [KS99]. In contrast to these logics where the logical connectives are adapted to talk about multiple truth values in our setting we are mainly interested in multiple truth values as diverse υ-pre-images of $\mathtt{T}$ and $\mathtt{F}$.

Example 1.5.6 (Failure of $\mathfrak{f}$ — $\mathcal{M}^{\beta\xi\mathfrak{b}} \in \mathfrak{M}_{\beta\xi\mathfrak{b}} \setminus \mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$)
We start by constructing a non-functional applicative structure by attaching distinguishing labels to functions without changing their applicative behavior. Let $\mathcal{B}$ be any set with $\mathtt{T} \in \mathcal{B}$ and $\mathtt{F} \notin \mathcal{B}$. Let $\mathcal{D}_o := \{\mathtt{F}\} \cup \mathcal{B}$ and $\mathcal{D}_\iota := \{*\}$ with $*$ as singleton element. For each function type $\alpha \to \beta$, let

$$\mathcal{D}_{\alpha\to\beta} := \{(i, f) \mid i \in \{0, 1\} \text{ and } f\colon \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta\}$$

Technically, we should write $\mathcal{D}^{\mathcal{B}}$ for $\mathcal{D}$, but to ease the notation, we wait until the model is defined to make its dependence on $\mathcal{B}$ explicit. We define application by $(i, f)@\mathsf{a} := f(\mathsf{a})$ whenever $(i, f) \in \mathcal{D}_{\alpha\to\beta}$ and $\mathsf{a} \in \mathcal{D}_\alpha$. It is easy to see that $(\mathcal{D}, @)$ is an applicative structure and is not functional. Consider, for example, the unique function $u\colon \mathcal{D}_\iota \longrightarrow \mathcal{D}_\iota$. For both $(0, u), (1, u) \in \mathcal{D}_{\iota\to\iota}$ we have $(i, u)@* \equiv *$, although $(0, u) \not\equiv (1, u)$.

We can define an evaluation function by induction on terms. We must begin by interpreting the constants. For the logical constants, let $\mathcal{E}(\neg) := (0, n)$ where $n(\mathsf{b}) := \mathtt{F}$ for every $\mathsf{b} \in \mathcal{B}$ and $n(\mathtt{F}) := \mathtt{T}$. Let $\mathcal{E}(\vee) := (0, d)$ where $d(\mathsf{b}) := (0, k^{\mathtt{T}})$ for every $\mathsf{b} \in \mathcal{B}$, $d(\mathtt{F}) := (0, id)$, $k^{\mathtt{T}}$ is the constant $\mathtt{T}$ function and id is the identity function from $\mathcal{D}_o$ to $\mathcal{D}_o$. For each type α, let $d(\Pi^\alpha) := (0, \pi^\alpha)$

where for each $(i,f) \in \mathcal{D}_{\alpha\to o}$, $\pi^\alpha((i,f)) := \mathtt{T}$ if $f(\mathsf{a}) \in \mathcal{B}$ for all $\mathsf{a} \in \mathcal{D}_\alpha$ and $\pi^\alpha(i,f) := \mathtt{F}$ otherwise. For each type α, let $\mathrm{q}^\alpha := (0, q^\alpha) \in \mathcal{D}_{\alpha\to\alpha\to o}$ where $q^\alpha(\mathsf{a}) := (0, s^{\mathsf{a}})$ and $s^{\mathsf{a}}(\mathsf{b}) := \mathtt{T}$ if $\mathsf{a} \equiv \mathsf{b}$ and $s^{\mathsf{a}}(\mathsf{b}) := \mathtt{F}$ otherwise. If primitive equality is present in the signature, let $\mathcal{E}(=^\alpha) := \mathrm{q}^\alpha$. Let $\mathcal{E}(w) \in \mathcal{D}_\alpha$ be arbitrary for parameters $w \in \Sigma_\alpha$.

For variables, we must define $\mathcal{E}_\varphi(X) := \varphi(X)$. Similarly, for application, we must define $\mathcal{E}_\varphi(\mathbf{FA}) := \mathcal{E}_\varphi(\mathbf{F})@\mathcal{E}_\varphi(\mathbf{A})$. For λ-abstractions, we have a choice. To be definite, we choose $\mathcal{E}_\varphi(\lambda X_\alpha.\mathbf{B}_\beta) := (0,f)$ where $f\colon \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$ is the function such that $f(\mathsf{a}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{B})$ for all $\mathsf{a} \in \mathcal{D}_\alpha$.

With some work (which we omit), one can show that this $\mathcal{E}$ is an evaluation function. Furthermore, taking v to be the function such that $v(\mathsf{b}) := \mathtt{T}$ for every $\mathsf{b} \in \mathcal{B}$ and $v(\mathtt{F}) := \mathtt{F}$, one can easily show that this is a valuation. Hence, $\mathcal{M}^{\mathcal{B}} := (\mathcal{D}, @, \mathcal{E}, v)$ is a Σ-model.

The objects q^α witness property $\mathfrak{q}$ for $\mathcal{M}^{\mathcal{B}}$ (and also show that this is a model with primitive equality, when primitive equality is in the signature). Note that the objects $(1, q^\alpha)$ also witness property $\mathfrak{q}$. So, in the non-functional case such witnesses are not unique.

We have already noted that property $\mathfrak{f}$ fails, since the applicative structure is not functional. One may question whether properties η or ξ hold. In fact, property η does not, as one may verify by computing, for example, $\mathcal{E}(\lambda F_{\alpha\to\beta}.F)$ and $\mathcal{E}(\lambda F_{\alpha\to\beta}X_\alpha.FX)$ for types α and β. We have $\mathcal{E}(\lambda F_{\alpha\to\beta}.F) \equiv (0, id)$ where id is the identity function from $\mathcal{D}_{\alpha\to\beta}$ to $\mathcal{D}_{\alpha\to\beta}$. However, $\mathcal{E}(\lambda F_{\alpha\to\beta}X_\alpha.FX) \equiv (0,p)$ where p is the function from $\mathcal{D}_{\alpha\to\beta}$ to $\mathcal{D}_{\alpha\to\beta}$ such that $p((i,f)) \equiv (0,f)$ for each $f\colon \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$. Property ξ does hold[6]. The reason is that if $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N})$ for every $\mathsf{a} \in \mathcal{D}_\alpha$, then $\mathcal{E}_\varphi(\lambda X_\alpha.\mathbf{M}) \equiv (0,f) \equiv \mathcal{E}_\varphi(\lambda X.\mathbf{N})$ where $f(\mathsf{a}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N})$ for every $\mathsf{a} \in \mathcal{D}_\alpha$.

Since $\mathcal{M}^{\mathcal{B}}$ is satisfies property $\mathfrak{q}$ but not property $\mathfrak{f}$, by Theorem 1.4.3(1) we have $\mathcal{M}^{\mathcal{B}} \not\models \mathrm{EXT}^{\alpha\to\beta}_{\doteq}$ for some types α and β. (One can easily check that, in fact, $\mathcal{M}^{\mathcal{B}} \not\models \mathrm{EXT}^{\alpha\to\beta}_{\doteq}$ for all types α and β by considering the witnesses $(0,f)$ and $(1,f)$ in $\mathcal{D}_{\alpha\to\beta}$ where $f\colon \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$ is any function.)

If $\mathcal{B} \equiv \{\mathtt{T}\}$, then the model $\mathcal{M}^{\beta\xi\mathfrak{b}} := \mathcal{M}^{\{\mathtt{T}\}}$ satisfies property $\mathfrak{b}$. So, we know $\mathcal{M}^{\beta\xi\mathfrak{b}} \in \mathfrak{M}_{\beta\xi\mathfrak{b}} \setminus \mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$. On the other hand, if b is any value with $\mathsf{b} \notin \{\mathtt{T},\mathtt{F}\}$, and $\mathcal{B} \equiv \{\mathtt{T},\mathsf{b}\}$, then the model $\mathcal{M}^{\beta\xi} := \mathcal{M}^{\{\mathtt{T},\mathsf{b}\}}$ does not satisfy property $\mathfrak{b}$. In this case, we know $\mathcal{M}^{\beta\xi} \in \mathfrak{M}_{\beta\xi} \setminus (\mathfrak{M}_{\beta\mathfrak{f}} \cup \mathfrak{M}_{\beta\xi\mathfrak{b}})$.

In Lemma 1.3.14, we have shown that $\beta\eta$-equality induces a functional congruence if the Σ_α is infinite for all types α. As a result, with such signatures, the term evaluation $\mathcal{TE}(\Sigma)^{\beta\eta}$ is functional (cf. Lemma 1.3.36). As noted in Remark 1.3.15, if Σ is finite, we cannot show that functionality holds. Nevertheless, even if Σ is finite, the evaluation $\mathcal{TE}(\Sigma)^{\beta\eta}$ interprets $\beta\eta$-convertible terms the same. We can use this idea to construct non-functional models which satisfy property η.

Example 1.5.7 (Failure of ξ — Instances of $\mathfrak{M}_\beta, \mathfrak{M}_{\beta\eta}, \mathfrak{M}_{\beta\mathfrak{b}}, \mathfrak{M}_{\beta\eta\mathfrak{b}}$)
Again, let $\mathcal{B}$ be any set with $\mathtt{T} \in \mathcal{B}$ and $\mathtt{F} \notin \mathcal{B}$. Choose constants $c_\iota, c_o \in \Sigma$ and let $\Sigma' := \{c_\iota, c_o\}$. By induction on types, we define $\mathbf{C}'_\alpha \in \mathit{cwff}_\alpha(\Sigma')\big\downarrow_{\beta\eta} \subseteq \mathit{cwff}_\alpha(\Sigma')\big\downarrow_\beta$. At base types, let $\mathbf{C}'_\iota := c_\iota$ and $\mathbf{C}'_o := c_o$. At function types, let $\mathbf{C}'_{\alpha\to\beta} := \lambda X_\alpha.\mathbf{C}'_\beta$. (Each $\mathbf{C}'_\alpha$ is of the form $\lambda\overline{X}.c_\beta$ where $\beta \in \{\iota, o\}$.) In particular, $\mathit{cwff}_\alpha(\Sigma')\big\downarrow_{\beta\eta}$ and $\mathit{cwff}_\alpha(\Sigma')\big\downarrow_\beta$ are non-empty for each type α.

We can now inductively define a map ρ from $\mathit{wff}_\alpha(\Sigma)$ to $\mathit{wff}_\alpha(\Sigma')$ which collapses terms to the smaller signature. For variables, let $\rho(X) := X$. For constants $w_\alpha \in \Sigma$ (including logical constants), let $\rho(w_\alpha) := \mathbf{C}'_\alpha$. For application and λ-abstraction, we simply define $\rho(\mathbf{FA}) := \rho(\mathbf{F})\rho(\mathbf{A})$ and $\rho(\lambda X.\mathbf{A}) := \lambda X.\rho(\mathbf{A})$. By induction on the formula $\mathbf{A}$, one can show $[\rho(\mathbf{B})/X]\rho(\mathbf{A}) \equiv$

[6]This construction is an example of how one constructs models for the simply typed λ-calculus using retractions. Such constructions will always yield models satisfying property ξ, but only yield models satisfying property η when each retraction is an isomorphism, in which case the applicative structure is functional.

$\rho([\mathbf{B}/X]\mathbf{A})$ for any $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, $\mathbf{B} \in \mathit{wff}_\beta(\Sigma)$ and X_β. From this, one can show $\rho(\mathbf{A}) \equiv_{\beta\eta} \rho(\mathbf{B})$ whenever $\mathbf{A} \equiv_{\beta\eta} \mathbf{B}$ for every $\mathbf{A}, \mathbf{B} \in \mathit{wff}_\alpha(\Sigma)$. Note also that $\rho(\mathbf{A}') \equiv \mathbf{A}'$ for every $\mathbf{A}' \in \mathit{wff}_\alpha(\Sigma')$.

We can construct a non-functional applicative structure using an indexing technique similar to Example 1.5.6. In this case, instead of indexing with $i \in \{0,1\}$, we use terms in $\mathit{cwff}_\alpha(\Sigma')\downarrow_*$ as indices. (Here $\mathbf{A}\downarrow_*$ means the β-normal form if $* \equiv \beta$ and the $\beta\eta$-normal form if $* \equiv \beta\eta$.) In essence, this index records some information about the "implementation" of the function. Note that $\mathit{cwff}_\iota(\Sigma')\downarrow_* \equiv \{c_\iota\}$ and $\mathit{cwff}_o(\Sigma')\downarrow_* \equiv \{c_o\}$. Let $\mathcal{D}_\iota := \{(c_\iota, 0)\}$ and $\mathcal{D}_o := \{(c_o, \mathtt{F})\} \cup \{(c_o, \mathsf{b}) \mid \mathsf{b} \in \mathcal{B}\}$. For function types, let $\mathcal{D}_{\alpha\to\beta}$ be the set of pairs $(\mathbf{F}'_{\alpha\to\beta}, f)$, where $\mathbf{F}' \in \mathit{cwff}_{\alpha\to\beta}(\Sigma')\downarrow_*$ and $f\colon \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$ is any function such that $\mathbf{B}' \equiv (\mathbf{F}'\mathbf{A}')\downarrow_*$ whenever $f(\mathbf{A}', a) \equiv (\mathbf{B}', b)$ for some value b. Application is defined as in Example 1.5.6: $(\mathbf{F}, f)@\mathsf{a} := f(\mathsf{a})$. The construction of this applicative structure closely follows Andrews' υ-complexes in [And71], except we have a very restricted signature Σ' which does not include logical constants.

To show that each domain is non-empty, we construct a particular element $\mathsf{c}^\alpha \in \mathcal{D}_\alpha$ for each type α. (This element will also be used to interpret parameters.) Let $\mathsf{c}^\iota := (c_\iota, 0)$, $\mathsf{c}^o := (c_o, \mathtt{F})$, and $\mathsf{c}^{\alpha\to\beta} := (\mathbf{C}'_{\alpha\to\beta}, k)$ where $k\colon \mathcal{D}_\alpha \longrightarrow \mathcal{D}_\beta$ is the constant function $k(\mathsf{a}) := \mathsf{c}^\beta$ for every $\mathsf{a} \in \mathcal{D}_\alpha$. The fact that $\mathsf{c}^{\alpha\to\beta} \in \mathcal{D}_{\alpha\to\beta}$ follows from $(\mathbf{C}'_{\alpha\to\beta}\mathbf{A})\downarrow_* \equiv \mathbf{C}'_\beta$.

One can see that the applicative structure is non-functional by noting $(\lambda X_\iota . X, f)$ and $(\lambda X_\iota . c_\iota, f)$ are distinct members of $\mathcal{D}_{\iota\to\iota}$, where f is the unique function taking $\mathcal{D}_\iota$ into itself. However, $(\lambda X_\iota . X, f)@\mathsf{c}^\iota \equiv \mathsf{c}^\iota \equiv (\lambda X_\iota . c_\iota, f)@\mathsf{c}^\iota$. In fact, once we define the evaluation function, this same example will show that property ξ will fail.

Let $\upsilon\colon \mathcal{D}_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$ be $\upsilon(\mathsf{c}^o) := \mathtt{F}$ and $\upsilon((c_o, \mathsf{b})) := \mathtt{T}$ for each $\mathsf{b} \in \mathcal{B}$. This will be the valuation function on the model.

We only sketch the definition of the evaluation function $\mathcal{E}$ and the proof that this gives a model $\mathcal{M}^{*,\mathcal{B}} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$. We can define $\mathcal{E}$ by induction on terms. First, we interpret parameters $w_\alpha \in \Sigma$ by $\mathcal{E}(w_\alpha) := \mathsf{c}^\alpha$. For logical constants $a_\alpha \in \Sigma$, we choose the first component of $\mathcal{E}(a_\alpha)$ to be $\mathbf{C}'_\alpha$ and the second component to be an appropriate function. We can define the witnesses q^α in a similar way and use these to interpret primitive equality, if it is present in the signature.

We are forced to let $\mathcal{E}_\varphi(X) := \varphi(X)$ and $\mathcal{E}_\varphi(\mathbf{FA}) := \mathcal{E}_\varphi(\mathbf{F})@\mathcal{E}_\varphi(\mathbf{A})$. For the λ-abstraction step, we choose $\mathcal{E}_\varphi(\lambda X_\alpha . \mathbf{B}_\beta) := ((\sigma(\rho(\lambda X . \mathbf{B})))\downarrow_*, f)$, where $f\colon \mathcal{D}_\alpha \to \mathcal{D}_\beta$ satisfies $f(\mathsf{a}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{B})$ for all $\mathsf{a} \in \mathcal{D}_\alpha$ and σ is the substitution defined by letting $\sigma(Y)$ be the first component of $\varphi(Y)$ for each $Y \in \mathit{free}(\lambda X . \mathbf{B})$. In order to show $\mathcal{E}$ is well-defined, one shows the first component of $\mathcal{E}_\varphi(\mathbf{A})$ is $(\sigma(\rho(\mathbf{A})))\downarrow_*$ (where σ is the substitution for $\mathit{free}(\mathbf{A})$ defined from the first components of the values of φ) for every formula $\mathbf{A}$.

The fact that $\mathcal{E}$ evaluates variables and application properly is immediate from the definition. The fact that $\mathcal{E}_\varphi(\mathbf{A})$ depends only the free variables in $\mathbf{A}$ follows by an induction on the definition of $\mathcal{E}$. To show $\mathcal{E}$ respects β-conversion if $* \equiv \beta$ and $\beta\eta$-conversion if $* \equiv \beta\eta$ (so that the model will also satisfy property η), one first shows $\mathcal{E}$ respects a single $\beta[\eta]$-reduction, then does an induction on the position of the redex, and finally does an induction on the number of $\beta[\eta]$-reductions.

Once these details are checked, we know $\mathcal{M}^{*,\mathcal{B}}$ is a model (with primitive equality, if present) satisfying property $\mathfrak{q}$. We already know the model will not satisfy property $\mathfrak{f}$ since the applicative structure is not functional. We can also check that the model will not satisfy property ξ by considering $\mathcal{E}(\lambda X_\iota . X)$ and $\mathcal{E}(\lambda X_\iota . c_\iota)$. We know $\mathcal{E}(\lambda X_\iota . X) \not\equiv \mathcal{E}(\lambda X_\iota . c_\iota)$ since the first components $((\lambda X_\iota . X)$ and $(\lambda X_\iota . c_\iota))$ are not equal. However, $\mathcal{D}_\iota$ has only one element, $\mathsf{c}^\iota(c_\iota, 0)$. So, we must have $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(X) \equiv \mathsf{c}^\iota \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(c_\iota)$ for every $\mathsf{a} \in \mathcal{D}_\iota$. This shows property ξ fails.

If $* \equiv \beta\eta$, then we have noted above that $\mathcal{E}$ respects $\beta\eta$-conversion. So, in this case, the model satisfies property η. If $* \equiv \beta$, then we can easily check $\mathcal{E}(\lambda F_{\iota\to\iota} X_\iota . FX) \not\equiv \mathcal{E}(\lambda F_{\iota\to\iota} . F)$ since the first components will differ. So, in this case, the model does not satisfy property η.

As in Example 1.5.6, if $\mathcal{B} \equiv \{\mathtt{T}\}$, then the models $\mathcal{M}^{\beta\mathfrak{b}} := \mathcal{M}^{\beta,\{\mathtt{T}\}}$ and $\mathcal{M}^{\beta\eta\mathfrak{b}} := \mathcal{M}^{\beta\eta,\{\mathtt{T}\}}$ satisfy property $\mathfrak{b}$. So, we know $\mathcal{M}^{\beta\mathfrak{b}} \in \mathfrak{M}_{\beta\mathfrak{b}} \setminus (\mathfrak{M}_{\beta\eta\mathfrak{b}} \cup \mathfrak{M}_{\beta\xi\mathfrak{b}})$ and $\mathcal{M}^{\beta\eta\mathfrak{b}} \in \mathfrak{M}_{\beta\eta\mathfrak{b}} \setminus \mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$. If $\mathcal{B} \equiv \{\mathtt{T}, \mathsf{b}\}$ where b is any value with $\mathsf{b} \notin \{\mathtt{T}, \mathtt{F}\}$, then the models $\mathcal{M}^{\beta} := \mathcal{M}^{\beta,\{\mathtt{T},\mathsf{b}\}}$ and $\mathcal{M}^{\beta\eta} := \mathcal{M}^{\beta\eta,\{\mathtt{T},\mathsf{b}\}}$ do not satisfy property $\mathfrak{b}$, so $\mathcal{M}^{\beta} \in \mathfrak{M}_{\beta} \setminus (\mathfrak{M}_{\beta\eta} \cup \mathfrak{M}_{\beta\xi} \cup \mathfrak{M}_{\beta\mathfrak{b}})$ and $\mathcal{M}^{\beta\eta} \in \mathfrak{M}_{\beta\eta} \setminus (\mathfrak{M}_{\beta\mathfrak{f}} \cup \mathfrak{M}_{\beta\eta\mathfrak{b}})$.

In particular, the models $\mathcal{M}^{\beta\eta}$ and $\mathcal{M}^{\beta\eta\mathfrak{b}}$ show that respecting η-conversion does not guarantee strong functional extensionality.

Thus we have given (sketches of) concrete models that distinguish model classes and shown that the inclusions between the $\mathfrak{M}_*$ model classes in Figure 1.1 are proper.

1.6 Model Existence

In this section we present the model existence theorems for the different semantical notions introduced in Section 1.3. The model existence theorems have the following form, where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$:

> **Theorem (Model Existence)**: *For a given abstract consistency class $\mathfrak{Acc}_*$ (cf. Definition 1.6.7) and a set $\Phi \in \mathfrak{Acc}_*$ there is a Σ-model $\mathcal{M}$ of Φ, such that $\mathcal{M} \in \mathfrak{M}_*$ (cf. Definition 1.3.49).*

The most important tools used in the proofs of the model existence theorems are the so-called Σ-Hintikka sets. These sets allow computations that resemble those in the considered semantical structures (e.g., Henkin models) and allow us to construct appropriate valuations for the term evaluation $\mathcal{TE}(\Sigma)^{\beta}$ defined in Definition 1.3.35.

The key step in the proof of the model existence theorems is an extension lemma, which guarantees a Σ-Hintikka set $\mathcal{H}$ for any sufficiently Σ-pure set of sentences Φ in $\mathfrak{A}_\Sigma$.

1.6.1 Abstract Consistency

Let us now review a few technicalities that we will need for the proofs of the model existence theorems.

Definition 1.6.1 (Compactness) Let $\mathcal{C}$ be a class of sets.

1. $\mathcal{C}$ is called *closed under subsets* if for any sets S and T, $S \in \mathcal{C}$ whenever $S \subseteq T$ and $T \in \mathcal{C}$.

2. $\mathcal{C}$ is called *compact* if for every set S we have $S \in \mathcal{C}$ iff every finite subset of S is a member of $\mathcal{C}$.

Lemma 1.6.2 *If $\mathcal{C}$ is compact, then $\mathcal{C}$ is closed under subsets.*

Proof: Suppose $S \subseteq T$ and $T \in \mathcal{C}$. Every finite subset A of S is a finite subset of T, and since $\mathcal{C}$ is compact we know that $A \in \mathcal{C}$. Thus $S \in \mathcal{C}$. $\square$

We will now introduce a technical side-condition that ensures that we always have enough witness constants.

Definition 1.6.3 (Sufficiently Σ-Pure) 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 $\mathit{wff}_\alpha(\Sigma)$, such that the elements of $\mathcal{P}$ do not occur in the sentences of Φ.

This can be obtained in practice by enriching the signature with spurious parameters. Another way would be to use specially marked variables (which may never be instantiated) as in [Koh94]. Note that for any set to be sufficiently Σ-pure, Σ_α must be infinite for each type α, since we have assumed that $\mathcal{V}_\alpha \subseteq \mathit{wff}(\Sigma)$ are infinite. Recall that in Remark 1.3.16 we assumed every Σ_α has a common (infinite) cardinality $\aleph_s$ for every type α. (One could easily show that no set of Σ-sentences could be sufficiently pure if, for example, Σ_ι is countable while $\Sigma_{\iota\to\iota}$ is uncountable. In such a case $\mathit{wff}_\alpha(\Sigma)$ is uncountable for every type α so one could not satisfy the sufficient purity condition at type ι.)

Notation 1.6.4 For reasons of legibility we will write $S * a$ for $S \cup \{a\}$, where S is a set. We will use this notation with the convention that $*$ associates to the left.

Definition 1.6.5 (Properties for Abstract Consistency Classes)
Let Γ_Σ be a class of sets of Σ-sentences and $\Phi \in \Gamma_\Sigma$. We define the following conditions, where $\alpha, \beta \in \mathcal{T}$, $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{F} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$, and $\mathbf{G}, \mathbf{H}, (\lambda X_\alpha.\mathbf{M}), (\lambda X_\alpha.\mathbf{N}) \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$.

∇_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 $\mathbf{A} \equiv_\beta \mathbf{B}$ and $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{B} \in \Gamma_\Sigma$.

∇_η If $\mathbf{A} \equiv_{\beta\eta} \mathbf{B}$ 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 $\mathbf{W} \in \mathit{cwff}_\alpha(\Sigma)$.

$\nabla_\exists$ 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.

$\nabla_{\mathfrak{b}}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$, then $\Phi * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$.

∇_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.

∇_{sat} Either $\Phi * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} \in \Gamma_\Sigma$.

For the optional case of primitive equality, i.e. when $=^\alpha \in \Sigma_{\alpha\to\alpha\to o}$ for all types α, we now add a set of further properties. While our first choice will be to combine the $\nabla_=^r$ property with $\nabla_=^{\doteq}$, we will later show that other pair combinations from this set are equivalent.

Definition 1.6.6 (Properties for Abstract Consistency Classes)
Suppose $=^\alpha \in \Sigma_{\alpha\to\alpha\to o}$ for all types α. Let Γ_Σ be a class of sets of Σ-sentences and $\Phi \in \Gamma_\Sigma$. We define for $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_\alpha(\Sigma)$ and $\mathbf{F} \in \mathit{cwff}_o(\Sigma)$ where $\mathbf{F}$ has a subterm of type α at position p:

$\nabla_=^r$ $\neg(\mathbf{A} =^\alpha \mathbf{A}) \notin \Phi$.

$\nabla^s_=$ If $\mathbf{F}[\mathbf{A}]_p \in \Phi$ and $\mathbf{A} =^\alpha \mathbf{B} \in \Phi$, then $\Phi * \mathbf{F}[\mathbf{B}]_p \in \Gamma_\Sigma$. [7]

$\nabla^{\dot{=}}_=$ If $\mathbf{A} =^\alpha \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} \doteq^\alpha \mathbf{B} \in \Gamma_\Sigma$.

$\nabla^{=}_{\dot{=}}$ If $\mathbf{A} \doteq^\alpha \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} =^\alpha \mathbf{B} \in \Gamma_\Sigma$.

$\nabla^{\dot{=}^-}_{=^-}$ If $\neg(\mathbf{A} =^\alpha \mathbf{B}) \in \Phi$, then $\Phi * \neg(\mathbf{A} \doteq^\alpha \mathbf{B}) \in \Gamma_\Sigma$.

$\nabla^{=^-}_{\dot{=}^-}$ If $\neg(\mathbf{A} \doteq^\alpha \mathbf{B}) \in \Phi$, then $\Phi * \neg(\mathbf{A} =^\alpha \mathbf{B}) \in \Gamma_\Sigma$.

Definition 1.6.7 (Abstract Consistency Classes) 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. Furthermore, when $=^\alpha \in \Sigma_{\alpha\to\alpha\to o}$ for all types α and the properties $\nabla^r_=$ and $\nabla^{\dot{=}}_=$ are valid then Γ_Σ is called an *abstract consistency class with primitive equality*. In the following we often simply use the phrase abstract consistency class to refer to an abstract consistency class with or without primitive equality. We will denote the collection of abstract consistency classes (with primitive equality) 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.

Remark 1.6.8 If primitive equality is not in the signature, $\mathfrak{Acc}_\beta$ corresponds to the abstract consistency property discussed by Andrews in [And71]. The only (technical) differences correspond to $\alpha\beta$-conversion. In [And71], α-conversion is handled in the ∇_β rule using α-standardized forms. Also, we have defined the ∇_β rule to work with β-conversion instead of β-reduction. We prefer this stronger version of ∇_β over the weaker option "If $\mathbf{A} \in \Phi$, then $\Phi * \mathbf{A}{\downarrow_\beta} \in \Gamma_\Sigma$" since it helps to avoid the use of ∇_{sat} in several proofs below. (Note that ∇_β follows from the weaker option and ∇_{sat}.) Furthermore, in practical applications, e.g. proving completeness of calculi, the stronger property is typically as easy to validate as the weaker one. An analogous argument applies to ∇_η.

Remark 1.6.9 While the work presented in this article is based on the choice of the primitive logical connectives $\neg, \vee$, and Π^α (and possibly primitive equality), 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 [Fit96]. It is clearly possible to achieve such a generalization for our framework as well. 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:

β-case: If $\beta \in \Phi$, then $\Phi * \beta_1 \in \Gamma_\Sigma$ or $\Phi * \beta_2 \in \Gamma_\Sigma$.

α-case: If $\alpha \in \Phi$, then $\Phi * \alpha_1 * \alpha_2 \in \Gamma_\Sigma$.

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 $\Phi * \delta w \in \Gamma_\Sigma$ for any parameter $w_\alpha \in \Sigma$ which does not occur in any sentence of Φ.

We often refer to property ∇_c as "atomic consistency". The next lemma shows that we also have the corresponding property for non-atoms.

[7] Although this resembles Lemma 1.3.23 which required property ξ, it is far weaker since $\mathbf{A}$ and $\mathbf{B}$ must be closed.

Lemma 1.6.10 (Non-atomic Consistency) *Let Γ_Σ be an abstract consistency class and $\mathbf{A} \in cwff_o(\Sigma)$, then for all $\Phi \in \Gamma_\Sigma$ we have $\mathbf{A} \notin \Phi$ or $\neg\mathbf{A} \notin \Phi$.*

Proof: (following a similar argument in [And71], Lemma 3.3.3)
If for some $\Phi \in \Gamma_\Sigma$ and $\mathbf{A} \in cwff_o(\Sigma)$ we have $\mathbf{A} \in \Phi$ and $\neg\mathbf{A} \in \Phi$, then $\{\mathbf{A}, \neg\mathbf{A}\} \in \Gamma_\Sigma$ since Γ_Σ is closed under subsets. Furthermore, using ∇_β and closure under subsets we can assume such an $\mathbf{A}$ is β-normal. We prove $\{\mathbf{A}, \neg\mathbf{A}\} \notin \Gamma_\Sigma$ for any β-normal $\mathbf{A} \in cwff_o(\Sigma)$ by induction on the number of logical constants in $\mathbf{A}$.

If $\mathbf{A}$ is atomic (which includes primitive equations), this follows immediately from ∇_c. Suppose $\mathbf{A} \equiv \neg\mathbf{B}$ for some $\mathbf{B} \in cwff_o(\Sigma)$ and $\{\neg\mathbf{B}, \neg\neg\mathbf{B}\} \in \Gamma_\Sigma$. By $\nabla_\neg$ and closure under subsets, we have $\{\neg\mathbf{B}, \mathbf{B}\} \in \Gamma_\Sigma$, contradicting the induction hypothesis for $\mathbf{B}$. Suppose $\mathbf{A} \equiv \mathbf{B} \vee \mathbf{C}$ for some $\mathbf{B}, \mathbf{C} \in cwff_o(\Sigma)$ and $\{\mathbf{B} \vee \mathbf{C}, \neg(\mathbf{B} \vee \mathbf{C})\} \in \Gamma_\Sigma$. By $\nabla_\vee$, $\nabla_\wedge$ and closure under subsets, we have either $\{\mathbf{B}, \neg\mathbf{B}\} \in \Gamma_\Sigma$ or $\{\mathbf{C}, \neg\mathbf{C}\} \in \Gamma_\Sigma$, contradicting the induction hypotheses for $\mathbf{B}$ and $\mathbf{C}$. Suppose $\mathbf{A} \equiv \Pi^\alpha\mathbf{B}$ for some $\mathbf{B}, \mathbf{C} \in cwff_{\alpha\to o}(\Sigma)$ and $\{\Pi^\alpha\mathbf{B}, \neg(\Pi^\alpha\mathbf{B})\} \in \Gamma_\Sigma$. Since Σ_α is assumed to be infinite (by Remark 1.3.16), there is a parameter $w_\alpha \in \Sigma_\alpha$ which does not occur in $\mathbf{A}$. Since w is a parameter, the sentence $\mathbf{B}w$ clearly has one less logical constant than $\Pi^\alpha\mathbf{B}$. However, we cannot directly apply the induction hypothesis as $\mathbf{B}w$ may not be β-normal. Since $\mathbf{B}$ is β-normal, the only way $\mathbf{B}w$ can fail to be β-normal is if $\mathbf{B}$ has the form $\lambda X_\alpha.\mathbf{C}$ for some $\mathbf{C} \in wff_o(\Sigma)$ where $free(\mathbf{C}) \subseteq \{X_\alpha\}$. In this case, it is easy to show that the reduct $[w/X]\mathbf{C}$ is β-normal and contains the same number of logical constants as $\mathbf{B}$. In either case, we can let $\mathbf{N}$ be the β-normal form of $\mathbf{B}w$ and apply the induction hypothesis to obtain $\{\mathbf{N}, \neg\mathbf{N}\} \notin \Gamma_\Sigma$. On the other hand, $\nabla_\exists$, $\nabla_\forall$, ∇_β and closure under subsets implies $\{\mathbf{N}, \neg\mathbf{N}\} \in \Gamma_\Sigma$, a contradiction. □

Remark 1.6.11 It is worth noting that the proof above depended upon the definition of $\nabla_\exists$. An alternative condition is "$\neg\Pi^\alpha\mathbf{F} \in \Phi$ implies $\Phi * \mathbf{FA} \in \Gamma_\Sigma$ for some $\mathbf{A} \in cwff_\alpha(\Sigma)$". Using this condition instead of $\nabla_\exists$ in the definition of abstract consistency class, we could not reduce non-atomic consistency to atomic consistency since we do not know $\mathbf{FA}$ has fewer logical constants than $\Pi^\alpha\mathbf{F}$. (There are other reasons why the alternative condition would be inappropriate.)

Remark 1.6.12 Note that for the connectives $\vee$ and Π^α there is a positive and a negative condition given in the definition above, namely $\nabla_\vee/\nabla_\wedge$ for $\vee$ and $\nabla_\forall/\nabla_\exists$ for Π^α. For $\doteq^o$ and $\doteq^{\alpha\to\beta}$ the situation is different since we need only conditions for the negative cases. Positive counterparts can be inferred by expanding the Leibniz definition of equality (cf. Lemma 1.6.13).

Lemma 1.6.13 (Leibniz Equality) *Let Γ_Σ be an abstract consistency class. The following properties are valid for all $\Phi \in \Gamma_\Sigma$, $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{C} \in cwff_\alpha(\Sigma)$ and $\mathbf{F}, \mathbf{G} \in cwff_{\alpha\to\beta}(\Sigma)$.*

$\nabla_{\doteq}^{r}$ $\neg(\mathbf{C} \doteq^\alpha \mathbf{C}) \notin \Phi$.

$\nabla_{\doteq}^{\to}$ *If $\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G} \in \Phi$, then $\Phi * \mathbf{FW} \doteq^\beta \mathbf{GW} \in \Gamma_\Sigma$ for any closed $\mathbf{W} \in cwff_\alpha(\Sigma)$.*

$\nabla_{\doteq}^{o}$ *If $\mathbf{A} \doteq^o \mathbf{B} \in \Phi$, then $\Phi * \mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$.*

Proof:

$\nabla_{\doteq}^{r}$ Assume $\neg(\mathbf{C} \doteq \mathbf{C}) \in \Phi$. By subset closure $\{\neg(\mathbf{C} \doteq \mathbf{C})\} \in \Gamma_\Sigma$ and by $\nabla_\exists$ with some parameter p which does not occur in $\mathbf{C}$ and ∇_β we get $\{\neg(\mathbf{C} \doteq \mathbf{C}), \neg(\neg p\mathbf{C} \vee p\mathbf{C})\} \in \Gamma_\Sigma$. The contradiction follows by $\nabla_\wedge$, $\nabla_\neg$ and ∇_c.

$\nabla_{\doteq}^{\to}$ Suppose $\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G} \in \Phi$. By application of $\nabla_\forall$ with $\lambda X_\alpha\mathbf{.FW} \doteq X\mathbf{W}$ and ∇_β we have $\Phi * (\neg(\mathbf{FW} \doteq \mathbf{FW}) \vee \mathbf{FW} \doteq \mathbf{GW}) \in \Gamma_\Sigma$. By $\nabla_\vee$ and subset closure we get $\Phi * \neg(\mathbf{FW} \doteq \mathbf{FW}) \in \Gamma_\Sigma$ or $\Phi * \mathbf{FW} \doteq \mathbf{GW} \in \Gamma_\Sigma$. The latter proves the assertion since the first option is ruled out by $\nabla_{\doteq}^{r}$.

$\nabla_{\doteq}^{o}$ Suppose $\mathbf{A} \doteq^{o} \mathbf{B} \in \Phi$. Applying $\nabla_\forall$ with $\lambda Y.Y$ we have $\Phi * (\lambda P_{o\to o}.\neg P\mathbf{A} \vee P\mathbf{B})(\lambda Y.Y) \in \Gamma_\Sigma$. By ∇_β and subset closure we get $\Phi * \neg\mathbf{A} \vee \mathbf{B} \in \Gamma_\Sigma$. Similarly, we further derive by $\nabla_\forall$ with $\lambda Y.\neg Y$, ∇_β, and subset closure that $\Phi * \neg\mathbf{A} \vee \mathbf{B} * \neg\neg\mathbf{A} \vee \neg\mathbf{B} \in \Gamma_\Sigma$. By applying $\nabla_\vee$ twice and subset closure we get the following four options: (i) $\Phi * \neg\mathbf{A} * \neg\neg\mathbf{A} \in \Gamma_\Sigma$, (ii) $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$, (iii) $\Phi * \mathbf{B} * \neg\neg\mathbf{A} \in \Gamma_\Sigma$, or (iv) $\Phi * \mathbf{B} * \neg\mathbf{B} \in \Gamma_\Sigma$. Cases (i) and (iv) are ruled out by non-atomic consistency. In case (iii) we furthermore get by $\nabla_\neg$ and subset closure that $\Phi * \mathbf{B} * \mathbf{A} \in \Gamma_\Sigma$. Thus, $\Phi * \neg\mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi * \mathbf{B} * \mathbf{A} \in \Gamma_\Sigma$. □

We could easily add respective properties for symmetry, transitivity, and congruence to the previous lemma. They can be shown analogously, i.e., they also follow from the properties of Leibniz equality.

In contrast to [And71], we work with saturated abstract consistency classes in order to simplify the proofs of the model existence theorems. For a discussion of the consequences of this decision, see Section 1.8.2.

Definition 1.6.14 (Saturatedness)
We call an abstract consistency class Γ_Σ saturated if it satisfies ∇_{sat}.

Remark 1.6.15 Clearly, not all abstract consistency classes are saturated, since the empty set is one that is not ($cwff_o(\Sigma)$ is certainly non-empty since $\forall P_o.P \in cwff_o(\Sigma)$).

Remark 1.6.16 The saturation condition ∇_{sat} can be very difficult to verify in practice. For example, showing that an abstract consistency class induced from a sequent calculus (as in [And71]) is saturated corresponds to showing cut-elimination (cf. [BBK02b]). Since Andrews [And71] did not use saturation, he could use his results to give a model-theoretic proof of cut-elimination for a sequent calculus. We cannot use the results of this article to obtain similar cut-elimination results.

We now investigate derived properties of primitive equality.

Lemma 1.6.17 (Primitive Equality) *Let Γ_Σ be an abstract consistency class with primitive equality, i.e., $=^\alpha \in \Sigma_{\alpha\to\alpha\to o}$ for all types $\alpha \in \mathcal{T}$, and $\nabla_{=}^{r}$ and $\nabla_{=}^{\doteq}$ hold. Then $\nabla_{\doteq}^{=}$ and $\nabla_{=}^{s}$ are valid. Furthermore, $\nabla_{=-}^{\doteq-}$ and $\nabla_{\doteq-}^{=-}$, are valid if Γ_Σ is saturated.*

Proof: To show $\nabla_{\doteq}^{=}$ we derive from $(\mathbf{A} \doteq^\alpha \mathbf{B}) \in \Phi$ by $\nabla_\forall$ with $\lambda X_\alpha.\mathbf{A} =^\alpha X$, ∇_β, and subset closure that $\Phi * \neg(\mathbf{A} = \mathbf{A}) \vee \mathbf{A} = \mathbf{B} \in \Gamma_\Sigma$. By $\nabla_\vee$ and subset closure we get $\Phi * \neg(\mathbf{A} = \mathbf{A}) \in \Gamma_\Sigma$ or $\Phi * \mathbf{A} = \mathbf{B} \in \Gamma_\Sigma$. The assertion follows from the latter option since the former is ruled out by $\nabla_{=}^{r}$.

In order to show $\nabla_{=}^{s}$ let $\mathbf{F}[\mathbf{A}]_p \in \Phi$, we derive from $\mathbf{A} =^\alpha \mathbf{B} \in \Phi$ by $\nabla_{=}^{\doteq}$ that $\Phi * (\mathbf{A} \doteq \mathbf{B}) \in \Gamma_\Sigma$. By $\nabla_\forall$ with $\lambda X.\mathbf{F}[X]_p$ (where $X \in \mathcal{V}_\alpha$ is not bound in $\mathbf{F}[\mathbf{A}]_p$), ∇_β, and subset closure we furthermore get that $\Phi * (\neg\mathbf{F}[\mathbf{A}]_p \vee \mathbf{F}[\mathbf{B}]_p) \in \Gamma_\Sigma$. Application of $\nabla_\vee$ and subset closure gives us $\Phi * \neg\mathbf{F}[\mathbf{A}]_p \in \Gamma_\Sigma$ or $\Phi * \mathbf{F}[\mathbf{B}]_p \in \Gamma_\Sigma$. The assertion follows from the latter option since the former is ruled out by $\mathbf{F}[\mathbf{A}]_p \in \Phi$ and non-atomic consistency.

The straightforward refutation proof for $\nabla_{\doteq-}^{=-}$ employs saturation, $\nabla_{=}^{\doteq}$, and non-atomic consistency. Similarly, the proof for $\nabla_{=-}^{\doteq-}$ employs saturation, $\nabla_{\doteq}^{=}$, and atomic consistency. □

The next theorem provides some alternatives to our choice of $\nabla^{\doteq}_{=}$ and $\nabla^{r}_{=}$ in the definition of abstract consistency classes with primitive equality provided that saturation holds. In practical applications the user may therefore choose the combination that suits best.

Theorem 1.6.18 (Alternative properties for primitive equality) *Let Γ_Σ be an abstract consistency class and let $=^\alpha \in \Sigma_{\alpha\to\alpha\to o}$ for all types $\alpha \in \mathcal{T}$. If Γ_Σ is saturated and validates one of the following combinations of properties, then it also validates $\nabla^{\doteq}_{=}$ and $\nabla^{r}_{=}$. The combinations are:*

1. $\nabla^{s}_{=}$ *and* $\nabla^{r}_{=}$.
2. $\nabla^{\doteq}_{=}$ *and* $\nabla^{=}_{\doteq}$.
3. $\nabla^{\doteq^-}_{=^-}$ *and* $\nabla^{=^-}_{\doteq^-}$.

Proof: To prove (1) we only have to show $\nabla^{\doteq}_{=}$. Let $(\mathbf{A} = \mathbf{B}) \in \Phi$ and suppose $\Phi * (\mathbf{A} \doteq \mathbf{B}) \notin \Gamma_\Sigma$. Then by saturation $\Phi * \neg(\mathbf{A} \doteq \mathbf{B}) \in \Gamma_\Sigma$ and by application of $\nabla^{s}_{=}$ we get a contradiction to $\nabla^{r}_{\doteq}$ (cf. Lemma 1.6.13).

To prove (2) we only have to show $\nabla^{r}_{=}$. Suppose $\neg(\mathbf{A} = \mathbf{A}) \in \Phi$. Since $\neg(\mathbf{A} \doteq \mathbf{A}) \notin \Phi$ by $\nabla^{r}_{\doteq}$ we get by saturation $\Phi * \mathbf{A} \doteq \mathbf{A} \in \Gamma_\Sigma$. The contradiction follows by $\nabla^{=}_{\doteq}$ and atomic consistency.

For (3) we first show $\nabla^{r}_{=}$. Suppose $\neg(\mathbf{A} = \mathbf{A}) \in \Phi$. Then by $\nabla^{\doteq^-}_{=^-}$ we get $\Phi * \neg(\mathbf{A} \doteq \mathbf{A}) \in \Gamma_\Sigma$ contradicting $\nabla^{r}_{\doteq}$. To show $\nabla^{\doteq}_{=}$ let $\mathbf{A} = \mathbf{B} \in \Phi$ and suppose $\Phi * \mathbf{A} \doteq \mathbf{B} \notin \Gamma_\Sigma$. By saturation we get $\Phi * \neg(\mathbf{A} \doteq \mathbf{B}) \in \Gamma_\Sigma$ and by application of $\nabla^{=^-}_{\doteq^-}$ we get contradiction to atomic consistency. □

Lemma 1.6.19 (Compactness of abstract consistency classes)
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 [And02], Proposition 2506)
We choose $\Gamma'_\Sigma := \{\Phi \subseteq \mathit{cwff}_o(\Sigma) \mid \text{every finite subset of } \Phi \text{ is in } \Gamma_\Sigma\}$. Now suppose that $\Phi \in \Gamma_\Sigma$. Γ_Σ is closed under subsets, so every finite subset of Φ is in Γ_Σ and thus $\Phi \in \Gamma'_\Sigma$. Hence $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Next let us show that each 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. Note that by Lemma 1.6.2 we have that Γ'_Σ is closed under subsets.

Next we show that if Γ_Σ satisfies ∇_*, then 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.

$\nabla_\neg$ 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 $\Phi * \mathbf{A} \in \Gamma'_\Sigma$.

$\nabla_\beta, \nabla_\eta, \nabla_\vee, \nabla_\wedge, \nabla_\forall, \nabla_\exists$ Analogous to $\nabla_\neg$.

∇_ξ Let $\Phi \in \Gamma'_\Sigma$, $\neg(\lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \Phi$ and Ψ be any finite subset of $\Phi * \neg([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N})$, where $w \in \Sigma_\alpha$ is a parameter that does not occur in Φ. We show that $\Psi \in \Gamma_\Sigma$.

Clearly $\Theta := (\Psi \setminus \{\neg([w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N})\}) * \neg(\lambda X.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N})$ is a finite subset of Φ and therefore $\Theta \in \Gamma_\Sigma$. Since Γ_Σ satisfies ∇_ξ and $\neg(\lambda X.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \Theta$, we have $\Theta * \neg([w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N}) \in \Gamma_\Sigma$. Furthermore, $\Psi \subseteq \Theta * \neg([w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N})$ and Γ_Σ is closed under subsets, so $\Psi \in \Gamma_\Sigma$. Thus every finite subset Ψ of $\Phi * \neg([w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N})$ is in Γ_Σ, and therefore by definition we have $\Phi * \neg([w/X]\mathbf{M} \doteq^{\alpha} [w/X]\mathbf{N}) \in \Gamma'_\Sigma$.

$\nabla_\mathfrak{f}$ Analogous to ∇_ξ.

$\nabla_\mathfrak{b}$ Let $\Phi \in \Gamma'_\Sigma$ with $\neg(\mathbf{A} \doteq \mathbf{B}) \in \Phi$. Assume $\Phi * \mathbf{A} * \neg\mathbf{B} \notin \Gamma_\Sigma$ and $\Phi * \neg\mathbf{A} * \mathbf{B} \notin \Gamma_\Sigma$. Then there exists finite subsets Φ_1 and Φ_2 of Φ, such that $\Phi_1 * \mathbf{A} * \neg\mathbf{B} \notin \Gamma_\Sigma$ and $\Phi_2 * \neg\mathbf{A} * \mathbf{B} \notin \Gamma_\Sigma$. Now we choose $\Phi_3 := \Phi_1 \cup \Phi_2 * \neg(\mathbf{A} \doteq \mathbf{B})$. Obviously Φ_3 is a finite subset of Φ and therefore $\Phi_3 \in \Gamma_\Sigma$. Since Γ_Σ satisfies $\nabla_\mathfrak{b}$, we have that $\Phi_3 * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi_3 * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$. From this and the fact that Γ_Σ is closed under subsets we get that $\Phi_1 * \mathbf{A} * \neg\mathbf{B} \in \Gamma_\Sigma$ or $\Phi_2 * \neg\mathbf{A} * \mathbf{B} \in \Gamma_\Sigma$, which contradicts our assumption.

∇_{sat} Let $\Phi \in \Gamma'_\Sigma$. Assume neither $\Phi * \mathbf{A}$ nor $\Phi * \neg\mathbf{A}$ is in Γ'_Σ. Then there are finite subsets Φ_1 and Φ_2 of Φ, such that $\Phi_1 * \mathbf{A} \notin \Gamma_\Sigma$ and $\Phi_2 * \neg\mathbf{A} \notin \Gamma_\Sigma$. As $\Psi := \Phi_1 \cup \Phi_2$ is a finite subset of Φ, we have $\Psi \in \Gamma_\Sigma$. Furthermore, $\Psi * \mathbf{A} \in \Gamma_\Sigma$ or $\Psi * \neg\mathbf{A} \in \Gamma_\Sigma$ because Γ_Σ is saturated. Γ_Σ is closed under subsets, so $\Phi_1 * \mathbf{A} \in \Gamma_\Sigma$ or $\Phi_2 * \neg\mathbf{A} \in \Gamma_\Sigma$. This is a contradiction, so we can conclude that if $\Phi \in \Gamma'_\Sigma$, then $\Phi * \mathbf{A} \in \Gamma'_\Sigma$ or $\Phi * \neg\mathbf{A} \in \Gamma'_\Sigma$.

In case primitive equality is present in the signature, we check the corresponding properties.

$\nabla^r_=$ Let $\Phi \in \Gamma'_\Sigma$ and assume $\neg(\mathbf{A} =^\alpha \mathbf{A}) \in \Phi$. $\{\neg(\mathbf{A} =^\alpha \mathbf{A})\}$ is clearly a finite subset of Φ and hence $\{\neg(\mathbf{A} =^\alpha \mathbf{A})\} \in \Gamma_\Sigma$ contradicting $\nabla^r_=$ in Γ_Σ.

$\nabla^{\doteq}_{=}, \nabla^{s}_{=}, \nabla^{=}_{\doteq}, \nabla^{\doteq^-}_{=^-}, \nabla^{=^-}_{\doteq^-}$ Analogous to $\nabla_\neg$. $\square$

1.6.2 Hintikka Sets

Hintikka sets connect syntax with semantics as they provide the basis for the model constructions in the model existence theorems. 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.

Definition 1.6.20 (Σ-Hintikka Properties) 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 $\neg\mathbf{A} \notin \mathcal{H}$.

$\vec{\nabla}_\neg$ If $\neg\neg\mathbf{A} \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$.

$\vec{\nabla}_\beta$ If $\mathbf{A} \in \mathcal{H}$ and $\mathbf{A} \equiv_\beta \mathbf{B}$, then $\mathbf{B} \in \mathcal{H}$.

$\vec{\nabla}_\eta$ If $\mathbf{A} \in \mathcal{H}$ and $\mathbf{A} \equiv_{\beta\eta} \mathbf{B}$, 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 $\mathbf{F}\mathbf{W} \in \mathcal{H}$ for each $\mathbf{W} \in cwff_\alpha(\Sigma)$.

$\vec{\nabla}_\exists$ If $\neg\Pi^\alpha\mathbf{F} \in \mathcal{H}$, then there is a parameter $w_\alpha \in \Sigma_\alpha$ such that $\neg(\mathbf{F}w) \in \mathcal{H}$.

$\vec{\nabla}_\mathfrak{b}$ If $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \mathcal{H}$, then $\{\mathbf{A}, \neg\mathbf{B}\} \subseteq \mathcal{H}$ or $\{\neg\mathbf{A}, \mathbf{B}\} \subseteq \mathcal{H}$.

$\vec{\nabla}_\xi$ 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 $\neg\mathbf{A} \in \mathcal{H}$.

$\vec{\nabla}^r_=$ $\neg(\mathbf{C} =^\alpha \mathbf{C}) \notin \mathcal{H}$.

$\vec{\nabla}^{\doteq}_=$ If $\mathbf{C} =^\alpha \mathbf{D} \in \mathcal{H}$, then $\mathbf{C} \doteq^\alpha \mathbf{D} \in \mathcal{H}$.

Definition 1.6.21 (Σ-Hintikka Set)
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 $\vec{\nabla}_\exists$. When primitive equality is present in the signature and $\mathcal{H}$ is a Hintikka set satisfying $\vec{\nabla}^r_=$ and $\vec{\nabla}^{\doteq}_=$ we call $\mathcal{H}$ a Σ-*Hintikka set with primitive equality.* We define the following collections of Hintikka sets (with primitive equality): $\mathfrak{Hint}_\beta$, $\mathfrak{Hint}_{\beta\eta}$, $\mathfrak{Hint}_{\beta\xi}$, $\mathfrak{Hint}_{\beta\mathfrak{f}}$, $\mathfrak{Hint}_{\beta\mathfrak{b}}$, $\mathfrak{Hint}_{\beta\eta\mathfrak{b}}$, $\mathfrak{Hint}_{\beta\xi\mathfrak{b}}$, and $\mathfrak{Hint}_{\beta\mathfrak{f}\mathfrak{b}}$, 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. If primitive equality is in the signature, we require $\mathcal{H} \in \mathfrak{Hint}_*$ to be a Hintikka set with primitive equality.

We will construct Hintikka sets as maximal elements of abstract consistency classes. To obtain a Hintikka set, we must explicitly show the property $\vec{\nabla}_\exists$ (and $\vec{\nabla}_\xi$ or $\vec{\nabla}_\mathfrak{f}$ when appropriate). This will ensure that Hintikka sets have enough parameters which act as witnesses.

Lemma 1.6.22 (Hintikka Lemma) *Let Γ_Σ be an abstract consistency class in $\mathfrak{Acc}_*$. Suppose a set $\mathcal{H} \in \Gamma_\Sigma$ satisfies the following properties:*

1. *$\mathcal{H}$ is subset-maximal in Γ_Σ (i.e., for each sentence $\mathbf{D} \in cwff_o(\Sigma)$ such that $\mathcal{H} * \mathbf{D} \in \Gamma_\Sigma$, we already have $\mathbf{D} \in \mathcal{H}$).*
2. *$\mathcal{H}$ satisfies $\vec{\nabla}_\exists$.*
3. *If $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$, then $\vec{\nabla}_\xi$ holds in $\mathcal{H}$.*
4. *If $* \in \{\beta\mathfrak{f}, \beta\mathfrak{f}\mathfrak{b}\}$, then $\vec{\nabla}_\mathfrak{f}$ holds in $\mathcal{H}$.*

Then, $\mathcal{H} \in \mathfrak{Hint}_*$. Furthermore, if Γ_Σ is saturated, then $\mathcal{H}$ satisfies $\vec{\nabla}_{sat}$.

Proof: $\mathcal{H}$ satisfies $\vec{\nabla}_\exists$ by assumption. Also, if $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$ ($* \in \{\beta\mathfrak{f}, \beta\mathfrak{f}\mathfrak{b}\}$), then we have explicitly assumed $\mathcal{H}$ satisfies $\vec{\nabla}_\xi$ ($\vec{\nabla}_\mathfrak{f}$). The fact that $\mathcal{H} \in \Gamma_\Sigma$ satisfies $\vec{\nabla}_c$ follows directly from non-atomic consistency (Lemma 1.6.10). Similarly, if primitive equality is in the signature, then $\mathcal{H}$ satisfies $\vec{\nabla}^r_=$ since $\mathcal{H} \in \Gamma_\Sigma$ and Γ_Σ satisfies $\nabla^r_=$. Every other $\vec{\nabla}_*$ property follows directly from the corresponding ∇_* property and maximality of $\mathcal{H}$ in Γ_Σ. For example, to show $\vec{\nabla}_\neg$, suppose $\neg\neg\mathbf{A} \in \mathcal{H}$. By $\nabla_\neg$, we know $\mathcal{H} * \mathbf{A} \in \Gamma_\Sigma$. By maximality of $\mathcal{H}$, we have $\mathbf{A} \in \mathcal{H}$. Checking $\vec{\nabla}_\beta$, $\vec{\nabla}_\eta$ (if $* \in \{\beta\eta, \beta\eta\mathfrak{b}\}$), $\vec{\nabla}_\wedge$, $\vec{\nabla}_\forall$, and $\vec{\nabla}^{\doteq}_=$ hold for $\mathcal{H}$ follows exactly this same pattern. Checking $\vec{\nabla}_\vee$, $\vec{\nabla}_\mathfrak{b}$ (if $* \in \{\beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$) and $\vec{\nabla}_{sat}$ (if Γ_Σ is saturated) follows a similar pattern, but with a simple case analysis. For example, to check $\vec{\nabla}_{sat}$, given $\mathbf{A} \in cwff_o(\Sigma)$, ∇_{sat} implies $\mathcal{H} * \mathbf{A} \in \Gamma_\Sigma$ or $\mathcal{H} * \neg\mathbf{A} \in \Gamma_\Sigma$. So, either $\mathbf{A} \in \mathcal{H}$ or $\neg\mathbf{A} \in \mathcal{H}$. □

It is worth noting that the converse of $\vec{\nabla}^{\doteq}_=$ also holds in Hintikka sets with primitive equality.

Lemma 1.6.23 *Suppose primitive equality is in the signature and $\mathcal{H}$ is a Hintikka set with primitive equality. Then, we have the following property for every type α and $\mathbf{A}, \mathbf{B} \in cwff_\alpha(\Sigma)$:*

$\overline{\nabla}^{\doteq}_{=}$ $\mathbf{A} =^\alpha \mathbf{B} \in \mathcal{H}$ *iff* $\mathbf{A} \doteq^\alpha \mathbf{B} \in \mathcal{H}$.

Proof: If $\mathbf{A} =^\alpha \mathbf{B} \in \mathcal{H}$, then $\mathbf{A} \doteq^\alpha \mathbf{B} \in \mathcal{H}$ by $\vec{\nabla}^{\doteq}_{=}$. For the converse direction assume that $\mathbf{A} \doteq^\alpha \mathbf{B} \in \mathcal{H}$. From this we get by $\vec{\nabla}_\forall$ with $\lambda X.\mathbf{A} = X$ and ∇_β that $\neg(\mathbf{A} = \mathbf{A}) \lor \mathbf{A} = \mathbf{B} \in \mathcal{H}$. Since $\neg(\mathbf{A} = \mathbf{A}) \notin \mathcal{H}$ by $\vec{\nabla}^r_{=}$, $\vec{\nabla}_\lor$ implies $\mathbf{A} =^\alpha \mathbf{B} \in \mathcal{H}$. ☐

It is helpful to note the following properties of Leibniz equality in Hintikka sets.

Lemma 1.6.24 *Suppose $\mathcal{H}$ is a Hintikka set. For any $\mathbf{F}, \mathbf{G} \in cwff_{\alpha\to\beta}(\Sigma)$ and $\mathbf{A}, \mathbf{B}, \mathbf{C} \in cwff_\alpha(\Sigma)$ (for types α and β), we have the following:*

$\vec{\nabla}^r_{\doteq}$ $\neg(\mathbf{A} \doteq^\alpha \mathbf{A}) \notin \mathcal{H}$.

$\vec{\nabla}^{tr}_{\doteq}$ *If $\mathbf{A} \doteq^\alpha \mathbf{B} \in \mathcal{H}$ and $\mathbf{B} \doteq^\alpha \mathbf{C} \in \mathcal{H}$, then $\mathbf{A} \doteq^\alpha \mathbf{C} \in \mathcal{H}$.*

$\vec{\nabla}^{\to}_{\doteq}$ *If $(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G}) \in \mathcal{H}$ and $(\mathbf{A} \doteq^\alpha \mathbf{B}) \in \mathcal{H}$, then $(\mathbf{FA} \doteq^\beta \mathbf{GB}) \in \mathcal{H}$.*

Proof:

$\vec{\nabla}^r_{\doteq}$ Suppose $\neg(\mathbf{A} \doteq^\alpha \mathbf{A}) \in \mathcal{H}$. By $\vec{\nabla}_\exists$ and $\vec{\nabla}_\beta$, there must be some parameter $q_{\alpha\to o}$ such that $\neg(\neg q\mathbf{A} \lor q\mathbf{A}) \in \mathcal{H}$. By $\vec{\nabla}_\land$, we have $\neg\neg q\mathbf{A} \in \mathcal{H}$ and $\neg q\mathbf{A} \in \mathcal{H}$, contradicting $\vec{\nabla}_c$.

$\vec{\nabla}^{tr}_{\doteq}$ Suppose $\mathbf{A} \doteq^\alpha \mathbf{B} \in \mathcal{H}$ and $\mathbf{B} \doteq^\alpha \mathbf{C} \in \mathcal{H}$. Let $\mathbf{Q}_{\alpha\to o}$ be the closed formula $(\lambda X_\alpha.\mathbf{A} \doteq^\alpha X)$. Applying $\vec{\nabla}_\forall$ to $\mathbf{B} \doteq^\alpha \mathbf{C} \in \mathcal{H}$ and $\mathbf{Q}$, we know $\neg(\mathbf{QB}) \lor \mathbf{QC} \in \mathcal{H}$. By $\vec{\nabla}_\lor$, we know $\neg(\mathbf{QB}) \in \mathcal{H}$ or $\mathbf{QC} \in \mathcal{H}$. If $\neg(\mathbf{QB}) \in \mathcal{H}$, then $\neg(\mathbf{A} \doteq^\alpha \mathbf{B}) \in \mathcal{H}$ by $\overline{\nabla}_\beta$, contradicting $\vec{\nabla}_c$. So, $\mathbf{QC} \in \mathcal{H}$ and hence $\mathbf{A} \doteq^\alpha \mathbf{C} \in \mathcal{H}$ as desired.

$\vec{\nabla}^{\to}_{\doteq}$ Let $\mathbf{P}_{(\alpha\to\beta)\to o}$ be the closed formula $(\lambda H_{\alpha\to\beta}.\mathbf{FA} \doteq^\beta H\mathbf{A})$, Applying $\vec{\nabla}_\forall$ to $(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G}) \in \mathcal{H}$ and $\mathbf{P}$, we have $\neg(\mathbf{PF}) \lor \mathbf{PG} \in \mathcal{H}$. By $\vec{\nabla}_\lor$, we know $\neg(\mathbf{PF}) \in \mathcal{H}$ or $\mathbf{PG} \in \mathcal{H}$. If $\neg(\mathbf{PF}) \in \mathcal{H}$, then $\neg(\mathbf{FA} \doteq^\beta \mathbf{FA}) \in \mathcal{H}$ by $\overline{\nabla}_\beta$, which contradicts $\vec{\nabla}^r_{\doteq}$. So, we must have $\mathbf{PG} \in \mathcal{H}$ and hence $(\mathbf{FA} \doteq^\beta \mathbf{GA}) \in \mathcal{H}$. Let $\mathbf{Q}_{\alpha\to o}$ be the closed formula $(\lambda X_\alpha.\mathbf{FA} \doteq^\beta \mathbf{G}X)$. Applying $\vec{\nabla}_\forall$ and $\vec{\nabla}_\lor$ to $(\mathbf{A} \doteq^\alpha \mathbf{B}) \in \mathcal{H}$, we know $\neg(\mathbf{QA}) \in \mathcal{H}$ or $\mathbf{QB} \in \mathcal{H}$. If $\neg(\mathbf{QA}) \in \mathcal{H}$, then $\neg(\mathbf{FA} \doteq^\beta \mathbf{GA}) \in \mathcal{H}$ by $\overline{\nabla}_\beta$, contradicting $\vec{\nabla}_c$. So, $\mathbf{QB} \in \mathcal{H}$ and hence $(\mathbf{FA} \doteq^\beta \mathbf{GB}) \in \mathcal{H}$ as desired.

Whenever a Hintikka set satisfies $\vec{\nabla}_{sat}$, we can prove far more closure properties. For example, we can prove converses of $\vec{\nabla}_\neg$, $\vec{\nabla}_\beta$, $\vec{\nabla}_\lor$, $\vec{\nabla}_\land$, $\vec{\nabla}_\forall$, $\vec{\nabla}_\exists$ and $\vec{\nabla}^{\doteq}_{=}$ (when primitive equality is in the signature). Also, if any of $\vec{\nabla}_\eta$, $\vec{\nabla}_\mathfrak{b}$, $\vec{\nabla}_\xi$ or $\vec{\nabla}_\mathfrak{f}$ hold, we can prove the corresponding converse. (We could call these properties $\overleftarrow{\nabla}_*$.) The proofs of the stronger properties $\overline{\nabla}_\neg$ and $\overline{\nabla}_\lor$ in Lemma 1.6.26 indicate how one would prove any of these converse properties.

Definition 1.6.25 (Saturated Set) We say a set of sentences $\mathcal{H}$ is *saturated* if it satisfies $\vec{\nabla}_{sat}$.

By Lemma 1.6.22, any Hintikka set constructed as a maximal member of a saturated abstract consistency class will be saturated. However, it is also possible for a maximal member of an abstract consistency class Γ_Σ to be saturated without Γ_Σ being saturated.

Lemma 1.6.26 (Saturated Sets Lemma) *Suppose $\mathcal{H}$ is a saturated Hintikka set. Then we have the following properties for every $\mathbf{A},\mathbf{B} \in cwff_o(\Sigma)$, $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma)$, and $\mathbf{C} \in cwff_\alpha(\Sigma)$ (for any type α):*

$\overline{\nabla}_\neg$ $\neg\mathbf{A} \in \mathcal{H}$ *iff* $\mathbf{A} \notin \mathcal{H}$.

$\overline{\nabla}_\vee$ $(\mathbf{A} \vee \mathbf{B}) \in \mathcal{H}$ *iff* $\mathbf{A} \in \mathcal{H}$ *or* $\mathbf{B} \in \mathcal{H}$.

$\overline{\nabla}_\forall$ $(\Pi^\alpha\mathbf{F}) \in \mathcal{H}$ *iff* $\mathbf{FD} \in \mathcal{H}$ *for every* $\mathbf{D} \in cwff_\alpha(\Sigma)$.

$\overline{\nabla}_\forall^\beta$ $(\Pi^\alpha\mathbf{F}) \in \mathcal{H}$ *iff* $(\mathbf{FD}){\downarrow_\beta} \in \mathcal{H}$ *for every* $\mathbf{D} \in cwff_\alpha(\Sigma){\downarrow_\beta}$.

$\overline{\nabla}_r$ $(\mathbf{C} \doteq^\alpha \mathbf{C}) \in \mathcal{H}$.

Proof:

$\overline{\nabla}_\neg$ If $\neg\mathbf{A} \in \mathcal{H}$, then $\mathbf{A} \notin \mathcal{H}$ by $\vec{\nabla}_c$. If $\mathbf{A} \notin \mathcal{H}$, then $\neg\mathbf{A} \in \mathcal{H}$ since $\mathcal{H}$ is saturated.

$\overline{\nabla}_\vee$ If $(\mathbf{A} \vee \mathbf{B}) \in \mathcal{H}$, then $\mathbf{A} \in \mathcal{H}$ or $\mathbf{B} \in \mathcal{H}$ by $\vec{\nabla}_\vee$. We prove the converse by contraposition. Suppose $(\mathbf{A} \vee \mathbf{B}) \notin \mathcal{H}$. By saturation we have $\neg(\mathbf{A} \vee \mathbf{B}) \in \mathcal{H}$, and by $\vec{\nabla}_\wedge$ we get $\neg\mathbf{A} \in \mathcal{H}$ and $\neg\mathbf{B} \in \mathcal{H}$. So, by $\vec{\nabla}_c$, $\mathbf{A} \notin \mathcal{H}$ and $\mathbf{B} \notin \mathcal{H}$.

$\overline{\nabla}_\forall$,$\overline{\nabla}_\forall^\beta$ One direction of $\overline{\nabla}_\forall$ is $\vec{\nabla}_\forall$. For $\overline{\nabla}_\forall^\beta$, note that if $(\Pi^\alpha\mathbf{F}) \in \mathcal{H}$, then for any $\mathbf{D} \in cwff_\alpha(\Sigma){\downarrow_\beta}$ we have $(\mathbf{FD}){\downarrow_\beta} \in \mathcal{H}$ by $\vec{\nabla}_\forall$ and $\vec{\nabla}_\beta$.

Suppose $(\Pi^\alpha\mathbf{F}) \notin \mathcal{H}$. By saturation, $\neg(\Pi^\alpha\mathbf{F}) \in \mathcal{H}$. By $\vec{\nabla}_\exists$, there is a parameter $w_\alpha \in \Sigma_\alpha$ such that $\neg(\mathbf{F}w) \in \mathcal{H}$. By $\vec{\nabla}_c$, we know $(\mathbf{F}w) \notin \mathcal{H}$. This shows the other direction of $\overline{\nabla}_\forall$. Furthermore, by $\vec{\nabla}_\beta$ we know $\neg(\mathbf{F}w){\downarrow_\beta} \in \mathcal{H}$ and so $(\mathbf{F}w){\downarrow_\beta} \notin \mathcal{H}$. Since w is β-normal, we also have the other direction of $\overline{\nabla}_\forall^\beta$.

$\overline{\nabla}_r$ This follows directly from saturation and $\vec{\nabla}_{\doteq}^r$. □

Lemma 1.6.27 (Saturated Sets Lemma for $\mathfrak{b}$) *Suppose $\mathcal{H} \in \mathfrak{Hint}_*$ where $* \in \{\beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathcal{H}$ is saturated, then the following property holds for all $\mathbf{A},\mathbf{B} \in cwff_o(\Sigma)$.*

$\overline{\nabla}_\mathfrak{b}$ $\mathbf{A} \doteq^o \mathbf{B} \in \mathcal{H}$ *or* $\mathbf{A} \doteq^o \neg\mathbf{B} \in \mathcal{H}$.

Proof: Suppose $(\mathbf{A} \doteq^o \mathbf{B}) \notin \mathcal{H}$ and $(\mathbf{A} \doteq^o \neg\mathbf{B}) \notin \mathcal{H}$. By saturation, $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \mathcal{H}$ and $\neg(\mathbf{A} \doteq^o \neg\mathbf{B}) \in \mathcal{H}$. By $\vec{\nabla}_\mathfrak{b}$, we must have $\{\mathbf{A}, \neg\mathbf{B}\} \subseteq \mathcal{H}$ or $\{\neg\mathbf{A}, \mathbf{B}\} \subseteq \mathcal{H}$. We must also have $\{\mathbf{A}, \neg\neg\mathbf{B}\} \subseteq \mathcal{H}$ or $\{\neg\mathbf{A}, \neg\mathbf{B}\} \subseteq \mathcal{H}$. Each of the four cases leads to an immediate contradiction to $\vec{\nabla}_c$. □

Lemma 1.6.28 (Saturated Sets Lemma for η) *Suppose $\mathcal{H} \in \mathfrak{Hint}_*$ where $* \in \{\beta\eta, \beta\eta\mathfrak{b}\}$. If $\mathcal{H}$ is saturated, then the following property holds for every type α and $\mathbf{A} \in cwff_\alpha(\Sigma)$:*

$\overline{\nabla}_\eta$ $(\mathbf{A} \doteq^\alpha \mathbf{A}{\downarrow_{\beta\eta}}) \in \mathcal{H}$.

Proof: If $(\mathbf{A} \doteq \mathbf{A}{\downarrow_{\beta\eta}}) \notin \mathcal{H}$, then by saturation $\neg(\mathbf{A} \doteq \mathbf{A}{\downarrow_{\beta\eta}}) \in \mathcal{H}$. So, by $\vec{\nabla}_\eta$ we have $\neg(\mathbf{A}{\downarrow_{\beta\eta}} \doteq^\alpha \mathbf{A}{\downarrow_{\beta\eta}}) \in \mathcal{H}$. But this contradicts $\vec{\nabla}_{\doteq}^r$. □

Lemma 1.6.29 (Saturated Sets Lemma for ξ) *Suppose $\mathcal{H} \in \mathfrak{Hint}_*$ where $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$. If $\mathcal{H}$ is saturated, then the following properties hold for all $\alpha, \beta \in \mathcal{T}$ and $(\lambda X_\alpha.\mathbf{M}), (\lambda X.\mathbf{N}) \in cwff_{\alpha\to\beta}(\Sigma)$:*

$\overline{\nabla}_\xi$ $\quad(\lambda X.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \mathcal{H}$ *iff* $([\mathbf{A}/X]\mathbf{M} \doteq^{\beta} [\mathbf{A}/X]\mathbf{N}) \in \mathcal{H}$ *for every* $\mathbf{A} \in cwff_\alpha(\Sigma)$.

$\overline{\nabla}_\xi^\beta$ $\quad(\lambda X.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \mathcal{H}$ *iff* $([\mathbf{A}/X]\mathbf{M} \doteq^{\beta} [\mathbf{A}/X]\mathbf{N})\big\downarrow_\beta \in \mathcal{H}$ *for every* $\mathbf{A} \in cwff_\alpha(\Sigma)\big\downarrow_\beta$.

Proof: Suppose $(\lambda X.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \mathcal{H}$ and $\mathbf{A} \in cwff_\alpha(\Sigma)$. We can apply $\vec{\nabla}_\forall$ and $\vec{\nabla}_\beta$ using the closed formula $(\lambda K_{\alpha\to\beta}.[\mathbf{A}/X]\mathbf{M} \doteq^{\beta} K\mathbf{A})$ to obtain

$$(\neg([\mathbf{A}/X]\mathbf{M} \doteq^{\beta} [\mathbf{A}/X]\mathbf{M}) \vee [\mathbf{A}/X]\mathbf{M} \doteq^{\beta} [\mathbf{A}/X]\mathbf{N}) \in \mathcal{H}$$

Since $\neg([\mathbf{A}/X]\mathbf{M} \doteq^{\beta} [\mathbf{A}/X]\mathbf{M}) \notin \mathcal{H}$ (by $\vec{\nabla}_{\doteq}^{r}$), we know $([\mathbf{A}/X]\mathbf{M} \doteq^{\beta} [\mathbf{A}/X]\mathbf{N}) \in \mathcal{H}$. This shows one direction of $\overline{\nabla}_\xi$. By $\vec{\nabla}_\beta$ we have $([\mathbf{A}/X]\mathbf{M} \doteq^{\beta} [\mathbf{A}/X]\mathbf{N})\big\downarrow_\beta \in \mathcal{H}$. Since this holds in particular for any $\mathbf{A} \in cwff_\alpha(\Sigma)\big\downarrow_\beta$, this shows one direction of $\overline{\nabla}_\xi^\beta$.

Suppose $(\lambda X.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \notin \mathcal{H}$. We show that there is a (β-normal) $\mathbf{A} \in cwff_\alpha(\Sigma)$ with $[\mathbf{A}/X]\mathbf{M} \doteq^{\beta} [\mathbf{A}/X]\mathbf{N} \notin \mathcal{H}$. By saturation, $\neg(\lambda X.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \mathcal{H}$. By $\vec{\nabla}_\xi$, there is a parameter $w_\alpha \in \Sigma_\alpha$ such that $\neg([w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N}) \in \mathcal{H}$. By $\vec{\nabla}_c$, $[w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N} \notin \mathcal{H}$. Choosing $\mathbf{A} := w$ we have the other direction of $\overline{\nabla}_\xi$. Since w is β-normal and $([w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N})\big\downarrow_\beta \notin \mathcal{H}$ (using $\vec{\nabla}_\beta$), we have the other direction of $\overline{\nabla}_\xi^\beta$. □

Lemma 1.6.30 (Saturated Sets Lemma for $\mathfrak{f}$) *Suppose $\mathcal{H} \in \mathfrak{Hint}_*$ where $* \in \{\beta\mathfrak{f}, \beta\mathfrak{fb}\}$. If $\mathcal{H}$ is saturated, then the following property holds for any types α and β and $\mathbf{G}, \mathbf{H} \in cwff_{\alpha\to\beta}(\Sigma)$.*

$\overline{\nabla}_\mathfrak{f}$ $\quad\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H} \in \mathcal{H}$ *iff* $\mathbf{GA} \doteq^{\beta} \mathbf{HA} \in \mathcal{H}$ *for every* $\mathbf{A} \in cwff_\alpha(\Sigma)$.

$\overline{\nabla}_\mathfrak{f}^\beta$ $\quad\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H} \in \mathcal{H}$ *iff* $(\mathbf{GA} \doteq^{\beta} \mathbf{HB})\big\downarrow_\beta \in \mathcal{H}$ *for every* $\mathbf{A}, \mathbf{B} \in cwff_\alpha(\Sigma)\big\downarrow_\beta$ *with* $(\mathbf{A} \doteq^{\alpha} \mathbf{B}) \in \mathcal{H}$.

Proof:

$\overline{\nabla}_\mathfrak{f}$ $\quad$ Let $(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \mathcal{H}$ and $\mathbf{A} \in cwff_\alpha(\Sigma)$. Since $(\mathbf{A} \doteq^{\alpha} \mathbf{A}) \in \mathcal{H}$ by $\overline{\nabla}_r$ we have $(\mathbf{GA} \doteq^{\beta} \mathbf{HB}) \in \mathcal{H}$ by $\vec{\nabla}_{\doteq}^{\to}$ (cf. Lemma 1.6.24). For the other direction, suppose $(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \notin \mathcal{H}$. By saturation, $\neg(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \mathcal{H}$. By $\vec{\nabla}_\mathfrak{f}$, there is a parameter $w_\alpha \in \Sigma$ such that $\neg(\mathbf{G}w \doteq^{\beta} \mathbf{H}w) \in \mathcal{H}$. So, $\mathbf{G}w \doteq^{\beta} \mathbf{H}w \notin \mathcal{H}$ by $\vec{\nabla}_c$ and we are done.

$\overline{\nabla}_\mathfrak{f}^\beta$ $\quad$ If $(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \mathcal{H}$ and $(\mathbf{A} \doteq^{\alpha} \mathbf{B}) \in \mathcal{H}$, then $(\mathbf{GA} \doteq^{\beta} \mathbf{HB}) \in \mathcal{H}$ by $\vec{\nabla}_{\doteq}^{\to}$, and hence $(\mathbf{GA} \doteq^{\beta} \mathbf{HB})\big\downarrow_\beta \in \mathcal{H}$ by $\vec{\nabla}_\beta$. For the other direction, suppose for every $\mathbf{A}, \mathbf{B} \in cwff_\alpha(\Sigma)\big\downarrow_\beta$ $(\mathbf{GA} \doteq^{\beta} \mathbf{HB})\big\downarrow_\beta \in \mathcal{H}$ whenever $(\mathbf{A} \doteq^{\alpha} \mathbf{B}) \in \mathcal{H}$. For any $\mathbf{A} \in cwff_\alpha(\Sigma)$, we know $(\mathbf{A} \doteq^{\alpha} \mathbf{A}) \in \mathcal{H}$ by $\overline{\nabla}_r$. By $\vec{\nabla}_\beta$ we know $(\mathbf{A}\big\downarrow_\beta \doteq^{\alpha} \mathbf{A}\big\downarrow_\beta) \in \mathcal{H}$. So, by our assumption, $(\mathbf{G}\, \mathbf{A}\big\downarrow_\beta \doteq^{\beta} \mathbf{H}\, \mathbf{A}\big\downarrow_\beta)\big\downarrow_\beta \in \mathcal{H}$. Using $\vec{\nabla}_\beta$ again, we have $(\mathbf{GA} \doteq^{\beta} \mathbf{HA}) \in \mathcal{H}$. Generalizing over $\mathbf{A}$, we have $(\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}) \in \mathcal{H}$ by $\overline{\nabla}_\mathfrak{f}$. □

In Lemma 1.3.22, we compared properties η, ξ and $\mathfrak{f}$ of models by showing $\mathfrak{f}$ is equivalent to η plus ξ. Similarly, Theorem 1.6.32 compares $\vec{\nabla}_\eta$, $\vec{\nabla}_\xi$, and $\vec{\nabla}_\mathfrak{f}$ as properties of Hintikka sets. Showing $\vec{\nabla}_\mathfrak{f}$ implies $\vec{\nabla}_\eta$ requires saturation and must be shown in several steps reflected by Lemma 1.6.31.

Lemma 1.6.31 *Let $\mathcal{H}$ be a saturated Hintikka set satisfying $\vec{\nabla}_\mathfrak{f}$.*

1. *For all $\mathbf{F} \in cwff_{\alpha\to\beta}(\Sigma)$ we have $(\lambda X_\alpha.\mathbf{F}X) \doteq^{\alpha\to\beta} \mathbf{F} \in \mathcal{H}$.*
2. *For all $\mathbf{A}, \mathbf{B} \in cwff_\alpha(\Sigma)$, if $\mathbf{A}$ η-reduces to $\mathbf{B}$ in one step, then $\mathbf{A} \doteq^\alpha \mathbf{B} \in \mathcal{H}$.*
3. *For all $\mathbf{A} \in cwff_\alpha(\Sigma)$, $\mathbf{A} \doteq^\alpha \mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$.*
4. *For all $\mathbf{A} \in cwff_o(\Sigma)$, if $\mathbf{A} \in \mathcal{H}$, then $\mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$.*

Proof:

1. Suppose $(\lambda X_\alpha.\mathbf{F}X) \doteq^{\alpha\to\beta} \mathbf{F} \notin \mathcal{H}$. By saturation, $\neg((\lambda X_\alpha.\mathbf{F}X) \doteq^{\alpha\to\beta} \mathbf{F}) \in \mathcal{H}$. By $\vec{\nabla}_\mathfrak{f}$, there is a parameter w_α such that $\neg(((\lambda X_\alpha.\mathbf{F}X)w) \doteq^\beta (\mathbf{F}w)) \in \mathcal{H}$. By $\vec{\nabla}_\beta$, $\neg((\mathbf{F}w) \doteq^\beta (\mathbf{F}w)) \in \mathcal{H}$, which contradicts $\vec{\nabla}^r_{\doteq}$ (cf. Lemma 1.6.24).

2. We prove this by induction on the position of the η-redex in $\mathbf{A}$. If $\mathbf{A}$ is the η-redex reduced to obtain $\mathbf{B}$, then this follows from part (1). Suppose $\mathbf{A} \equiv (\mathbf{F}_{\gamma\to\alpha}\mathbf{C}_\gamma)$ and $\mathbf{B} \equiv (\mathbf{G}_{\gamma\to\alpha}\mathbf{C})$ where $\mathbf{F}$ η-reduces to $\mathbf{G}$ in one step. By induction, we know $\mathbf{F} \doteq^{\gamma\to\alpha} \mathbf{G} \in \mathcal{H}$. By $\overline{\nabla}_r$, $\mathbf{C} \doteq^\gamma \mathbf{C} \in \mathcal{H}$. By $\vec{\nabla}^{\to}_{\doteq}$, we have $(\mathbf{FC}) \doteq^\alpha (\mathbf{GC}) \in \mathcal{H}$ as desired. The case in which $\mathbf{A} \equiv (\mathbf{F}_{\gamma\to\alpha}\mathbf{C}_\gamma)$ and $\mathbf{B} \equiv (\mathbf{FD}_\gamma)$ where $\mathbf{C}$ η-reduces to $\mathbf{D}$ in one step is analogous.

   Finally, suppose $\mathbf{A} \equiv (\lambda Y_\beta.\mathbf{C}_\gamma)$ and $\mathbf{B} \equiv (\lambda Y_\beta.\mathbf{D}_\gamma)$ where $\mathbf{C}$ η-reduces to $\mathbf{D}$ in one step. Let p be the position of the redex in $\mathbf{C}$. Assume $\mathbf{A} \doteq^{\beta\to\gamma} \mathbf{B} \notin \mathcal{H}$. By saturation, $\neg(\mathbf{A} \doteq^{\beta\to\gamma} \mathbf{B}) \in \mathcal{H}$. By $\vec{\nabla}_\mathfrak{f}$, there is some parameter w_β such that $\neg(\mathbf{A}w \doteq^\gamma \mathbf{B}w) \in \mathcal{H}$. By $\vec{\nabla}_\beta$, we know $\neg([w/Y]\mathbf{C} \doteq^\gamma [w/Y]\mathbf{D}) \in \mathcal{H}$. Note that $[w/Y]\mathbf{C}$ η-reduces to $[w/Y]\mathbf{D}$ in one step by reducing the redex at position p in $[w/Y]\mathbf{C}$. So, by the induction hypothesis, $[w/Y]\mathbf{C} \doteq^\gamma [w/Y]\mathbf{D} \in \mathcal{H}$, contradicting $\vec{\nabla}_c$.

3. This follows by induction on the number of $\beta\eta$-reductions from $\mathbf{A}$ to $\mathbf{A}{\downarrow_{\beta\eta}}$. If $\mathbf{A}$ is $\beta\eta$-normal, we have $\mathbf{A} \doteq^\alpha \mathbf{A} \in \mathcal{H}$ by $\overline{\nabla}_r$. If $\mathbf{A}$ reduces to $\mathbf{A}{\downarrow_{\beta\eta}}$ in $n+1$ steps, then there is some $\mathbf{B}_\alpha$ such that $\mathbf{A}$ reduces to $\mathbf{B}$ in one step and $\mathbf{B}$ reduces to $\mathbf{A}{\downarrow_{\beta\eta}}$ in n steps. By induction, we have $\mathbf{B} \doteq^\alpha \mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$. If $\mathbf{A}$ β-reduces to $\mathbf{B}$ in one step, then $\mathbf{A} \doteq^\alpha \mathbf{B} \in \mathcal{H}$ by $\overline{\nabla}_r$ and $\vec{\nabla}_\beta$. If $\mathbf{A}$ η-reduces to $\mathbf{B}$ in one step, then $\mathbf{A} \doteq^\alpha \mathbf{B} \in \mathcal{H}$ by part (2). Using $\vec{\nabla}^{tr}_{\doteq}$, $\mathbf{A} \doteq^\alpha \mathbf{B} \in \mathcal{H}$ and $\mathbf{B} \doteq^\alpha \mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$ imply $\mathbf{A} \doteq^\alpha \mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$ as desired.

4. Suppose $\mathbf{A} \in \mathcal{H}$. By part (3), $\mathbf{A} \doteq^o \mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$. By $\vec{\nabla}_\forall$, $\neg(\lambda X_o.X)\mathbf{A} \vee (\lambda X_o.X)\, \mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$. By $\vec{\nabla}_\beta$ and $\vec{\nabla}_\vee$, we have $\neg\mathbf{A} \in \mathcal{H}$ (contradicting $\vec{\nabla}_c$) or $\mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$. Hence, $\mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$. □

Theorem 1.6.32 *Let $\mathcal{H}$ be a Hintikka set.*

1. *If $\mathcal{H}$ satisfies $\vec{\nabla}_\eta$ and $\vec{\nabla}_\xi$, then $\mathcal{H}$ satisfies $\overline{\nabla}_\mathfrak{f}$.*
2. *If $\mathcal{H}$ satisfies $\vec{\nabla}_\mathfrak{f}$, then $\mathcal{H}$ satisfies $\vec{\nabla}_\xi$.*
3. *If $\mathcal{H}$ is saturated and satisfies $\vec{\nabla}_\mathfrak{f}$, then $\mathcal{H}$ satisfies $\vec{\nabla}_\eta$.*

Proof:

1. Suppose $\neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G}) \in \mathcal{H}$. By $\vec{\nabla}_\eta$, $\neg((\lambda X_\alpha.\mathbf{F}X) \doteq^{\alpha\to\beta} (\lambda X.\mathbf{G}X)) \in \mathcal{H}$. By $\vec{\nabla}_\xi$, there is a parameter w_α such that $\neg((\mathbf{F}w) \doteq^{\beta} (\mathbf{G}w)) \in \mathcal{H}$. Thus, $\vec{\nabla}_\mathfrak{f}$ holds.

2. Suppose $\neg(\lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N}) \in \mathcal{H}$. By $\vec{\nabla}_\mathfrak{f}$, there is a parameter w_α such that $\neg((\lambda X_\alpha.\mathbf{M})w \doteq^{\beta} (\lambda X.\mathbf{N})w) \in \mathcal{H}$. By $\vec{\nabla}_\beta$, we have $\neg([w/X]\mathbf{M} \doteq^{\beta} [w/X]\mathbf{N}) \in \mathcal{H}$. Thus, $\vec{\nabla}_\xi$ holds.

3. Suppose $\mathbf{A} \in \mathcal{H}$, $\mathbf{B} \in \mathit{cwff}_o(\Sigma)$ and $\mathbf{A}\equiv_{\beta\eta}\mathbf{B}$. Assume $\mathbf{B} \notin \mathcal{H}$. By saturation, we know $\neg\mathbf{B} \in \mathcal{H}$. By Lemma 1.6.31(4), we know $\mathbf{A}{\downarrow_{\beta\eta}} \in \mathcal{H}$ and $\neg\,\mathbf{B}{\downarrow_{\beta\eta}} \in \mathcal{H}$. Since $\mathbf{A}{\downarrow_{\beta\eta}} \equiv \mathbf{B}{\downarrow_{\beta\eta}}$, this contradicts $\vec{\nabla}_c$. □

1.6.3 Model Existence Theorems

We shall now present the proof of the abstract extension lemma, which will nearly immediately yield the model existence theorems. For the proof we adapt the construction of Henkin's completeness proof from [Hen50, Hen96].

Lemma 1.6.33 (Abstract Extension Lemma) *Let Σ be a signature, Γ_Σ be a compact abstract consistency class in $\mathfrak{Acc}_*$, where $* \in \{\beta,\beta\eta,\beta\xi,\beta\mathfrak{b},\beta\eta\mathfrak{b},\beta\xi\mathfrak{b},\beta\mathfrak{f},\beta\mathfrak{fb}\}$ (cf. Definitions 1.3.49 and 1.3.50) and let $\Phi \in \Gamma_\Sigma$ be sufficiently Σ-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.*

In the following argument, note that α, β, and γ are types as usual, while δ, ϵ, σ and τ are ordinals. **Proof:** By Remark 1.3.16, there is an infinite cardinal $\aleph_s$ which is the cardinality of Σ_α for each type α. This easily implies $\mathit{cwff}_\alpha(\Sigma)$ is of cardinality $\aleph_s$ for each type α. Let ϵ be the first ordinal of this cardinality. (In the countable case, ϵ is ω.) Since the cardinality of $\mathit{cwff}_o(\Sigma)$ is $\aleph_s$, we can use the well-ordering principle to enumerate $\mathit{cwff}_o(\Sigma)$ as $(\mathbf{A}^\delta)_{\delta<\epsilon}$.

Let α be a type. For each $\delta < \epsilon$, let U^δ_α be the set of constants which occur in a sentence in the set $\{\mathbf{A}^\sigma \mid \sigma \leq \delta\}$. Since $\delta < \epsilon$, the set $\{\mathbf{A}^\sigma \mid \sigma \leq \delta\}$ has cardinality less than $\aleph_s$. Hence, U^δ_α has cardinality less than $\aleph_s$. By sufficient purity, we know there is a set of parameters $P_\alpha \subseteq \Sigma_\alpha$ of cardinality $\aleph_s$ such that the parameters in P_α do not occur in Φ. Since, considering cardinality, we cannot have $P_\alpha \subseteq U^\delta_\alpha$ for any $\delta < \epsilon$, we know $P_\alpha \setminus U^\delta_\alpha$ is non-empty for each $\delta < \epsilon$. Using the axiom of choice, we can find a sequence $(w^\delta_\alpha)_{\delta<\epsilon}$ where for each $\delta < \epsilon$, $w^\delta_\alpha \in P_\alpha \setminus U^\delta_\alpha$. That is, for each type α, we know w^δ_α is a parameter of type α which does not occur in $\Phi \cup \{\mathbf{A}^\sigma \mid \sigma \leq \delta\}$. As a consequence, if w^δ_α occurs in $\mathbf{A}^\sigma$, then $\delta < \sigma$.

The parameters w^δ_α are intended to serve as witnesses. To ease the argument, we define two sequences of witnessing sentences related to the sequence $(\mathbf{A}^\delta)_{\delta<\epsilon}$. For each $\delta < \epsilon$, let $\mathbf{E}^\delta := \neg(\mathbf{B}w^\delta_\alpha)$ if $\mathbf{A}^\delta$ is of the form $\neg(\Pi^\alpha\mathbf{B})$, and let $\mathbf{E}^\delta := \mathbf{A}^\delta$ otherwise. If $* \in \{\beta\mathfrak{f},\beta\mathfrak{fb}\}$ and $\mathbf{A}^\delta$ is of the form $\neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G})$, let $\mathbf{X}^\delta := \neg(\mathbf{F}w^\delta_\alpha \doteq^{\beta} \mathbf{G}w^\delta_\alpha)$. If $* \in \{\beta\xi,\beta\xi\mathfrak{b}\}$ and $\mathbf{A}^\delta$ is of the form $\neg((\lambda X_\alpha.\mathbf{M}) \doteq^{\alpha\to\beta} (\lambda X.\mathbf{N}))$, let $\mathbf{X}^\delta := \neg([w^\delta_\alpha/X]\mathbf{M} \doteq^{\beta} [w^\delta_\alpha/X]\mathbf{N})$. Otherwise, let $\mathbf{X}^\delta := \mathbf{A}^\delta$. (Notice that any sentence $\neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G})$ is also of the form $\neg(\Pi^\gamma\mathbf{B})$, where γ is $(\alpha \to \beta) \to o$. So, whenever $\mathbf{X}^\delta \not\equiv \mathbf{A}^\delta$, we must also have $\mathbf{E}^\delta \not\equiv \mathbf{A}^\delta$.)

We construct $\mathcal{H}$ by inductively constructing a transfinite sequence $(\mathcal{H}^\delta)_{\delta<\epsilon}$ such that $\mathcal{H}^\delta \in \Gamma_\Sigma$ for each $\delta < \epsilon$. Then the Σ-Hintikka set is $\mathcal{H} := \bigcup_{\delta<\epsilon}\mathcal{H}^\delta$. We define $\mathcal{H}^0 := \Phi$. For limit ordinals δ, we define $\mathcal{H}^\delta := \bigcup_{\sigma<\delta}\mathcal{H}^\sigma$.

In the successor case, if $\mathcal{H}^\delta * \mathbf{A}^\delta \in \Gamma_\Sigma$, then we let $\mathcal{H}^{\delta+1} := \mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta * \mathbf{X}^\delta$. If $\mathcal{H}^\delta * \mathbf{A}^\delta \notin \Gamma_\Sigma$, we let $\mathcal{H}^{\delta+1} := \mathcal{H}^\delta$.

We show by induction that for every $\delta < \epsilon$, type α and parameter w_α^τ which occurs in some sentence in $\mathcal{H}^\delta$, we have $\tau < \delta$. The base case holds since no w_α^τ occurs in any sentence in $\mathcal{H}^0 \equiv \Phi$. For any limit ordinal δ, if w_α^τ occurs in some sentence in $\mathcal{H}^\delta$, then by definition of $\mathcal{H}^\delta$, w_α^τ already occurs in some sentence in $\mathcal{H}^\sigma$ for some $\sigma < \delta$. So, $\tau < \sigma < \delta$.

For any successor ordinal $\delta + 1$, suppose w_α^τ occurs in some sentence in $\mathcal{H}^{\delta+1}$. If it already occurred in a sentence in $\mathcal{H}^\delta$, then we have $\tau < \delta < \delta + 1$ by the inductive assumption. So, we need only consider the case where w_α^τ occurs in a sentence in $\mathcal{H}^{\delta+1} \setminus \mathcal{H}^\delta$. Note that $(\mathcal{H}^{\delta+1} \setminus \mathcal{H}^\delta) \subseteq \{\mathbf{A}^\delta, \mathbf{E}^\delta, \mathbf{X}^\delta\}$. In any case, note that if τ is δ, then we are done, since $\delta < \delta + 1$. If w_α^τ is any parameter with $\tau \not\equiv \delta$ and occurs in $\mathbf{E}^\delta$ or $\mathbf{X}^\delta$, then it must also occur in $\mathbf{A}^\delta$ (by inspecting the possible definitions of $\mathbf{E}^\delta$ and $\mathbf{X}^\delta$), in which case $\tau < \delta < \delta + 1$.

In particular, we now know w_α^δ does not occur in any sentence in $\mathcal{H}^\delta$ for any $\delta < \epsilon$ and type α.

Next we show by induction that $\mathcal{H}^\delta \in \Gamma_\Sigma$ for all $\delta < \epsilon$. The base case holds by the assumption that $\mathcal{H}^0 \equiv \Phi \in \Gamma_\Sigma$. For any limit ordinal δ, assume $\mathcal{H}^\sigma \in \Gamma_\Sigma$ for every $\sigma < \delta$. We have $\mathcal{H}^\delta \equiv \bigcup_{\sigma<\delta} \mathcal{H}^\sigma \in \Gamma_\Sigma$ by compactness, since any finite subset of $\mathcal{H}^\delta$ is a subset of $\mathcal{H}^\sigma$ for some $\sigma < \delta$.

For any successor ordinal $\delta + 1$, we assume $\mathcal{H}^\delta \in \Gamma_\Sigma$. We have to show that $\mathcal{H}^{\delta+1} \in \Gamma_\Sigma$. This is trivial in case $\mathcal{H}^\delta * \mathbf{A}^\delta \notin \Gamma_\Sigma$ (for all abstract consistency classes) since $\mathcal{H}^{\delta+1} \equiv \mathcal{H}^\delta$. Suppose $\mathcal{H}^\delta * \mathbf{A}^\delta \in \Gamma_\Sigma$. We consider three sub-cases:

1. If $\mathbf{E}^\delta \equiv \mathbf{A}^\delta$ and $\mathbf{X}^\delta \equiv \mathbf{A}^\delta$, then $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta * \mathbf{X}^\delta \in \Gamma_\Sigma$ since $\mathcal{H}^\delta * \mathbf{A}^\delta \in \Gamma_\Sigma$.

2. If $\mathbf{E}^\delta \not\equiv \mathbf{A}^\delta$ and $\mathbf{X}^\delta \equiv \mathbf{A}^\delta$, then $\mathbf{A}^\delta$ is of the form $\neg\Pi^\alpha\mathbf{B}$ and $\mathbf{E}^\delta \equiv \neg\mathbf{B}w_\alpha^\delta$. We conclude that $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta \in \Gamma_\Sigma$ by $\nabla_\exists$ since w_α^δ does not occur in $\mathbf{A}^\delta$ or any sentence in $\mathcal{H}^\delta$. Since $\mathbf{X}^\delta \equiv \mathbf{A}^\delta$, this is the same as concluding $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta * \mathbf{X}^\delta \in \Gamma_\Sigma$.

3. If $\mathbf{X}^\delta \not\equiv \mathbf{A}^\delta$, then $* \in \{\beta\xi, \beta\mathfrak{f}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ (by the definition of $\mathbf{X}^\delta$). $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta \in \Gamma_\Sigma$ by $\nabla_\exists$ since $w_{(\alpha\to\beta)\to o}^\delta$ does not occur in $\mathbf{A}^\delta$ or any sentence in $\mathcal{H}^\delta$. Now, w_α^δ (which is different from $w_{(\alpha\to\beta)\to o}^\delta$ since it has a different type) does not occur in any sentence in $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta$. We have $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta * \mathbf{X}^\delta \in \mathcal{H}$ by ∇_ξ (if $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$) or by $\nabla_\mathfrak{f}$ (if $* \in \{\beta\mathfrak{f}, \beta\mathfrak{f}\mathfrak{b}\}$).

Since Γ_Σ is compact, we also have $\mathcal{H} \in \Gamma_\Sigma$.

Now we know that our inductively defined set $\mathcal{H}$ is indeed in Γ_Σ and that $\Phi \subseteq \mathcal{H}$. In order to apply Lemma 1.6.22, we must check $\mathcal{H}$ is maximal, satisfies $\vec{\nabla}_\exists$, $\vec{\nabla}_\xi$ (if $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$), and $\vec{\nabla}_\mathfrak{f}$ (if $* \in \{\beta\mathfrak{f}, \beta\mathfrak{f}\mathfrak{b}\}$). It is immediate from the construction that $\vec{\nabla}_\exists$ holds since if $\neg(\Pi^\alpha\mathbf{F}) \in \mathcal{H}$, then $\neg(\mathbf{F}w_\alpha^\delta) \in \mathcal{H}$ where δ is the ordinal such that $\mathbf{A}^\delta \equiv \neg(\Pi^\alpha\mathbf{F})$. If $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$, then we have ensured $\vec{\nabla}_\xi$ holds since $\neg([w_\alpha^\delta/X]\mathbf{M} \doteq^\beta [w_\alpha^\delta/X]\mathbf{N}) \in \mathcal{H}$ whenever $\neg((\lambda X_\alpha.\mathbf{M}) \doteq^{\alpha\to\beta} (\lambda X.\mathbf{N})) \in \mathcal{H}$ where δ is the ordinal such that $\mathbf{A}^\delta \equiv \neg((\lambda X_\alpha.\mathbf{M}) \doteq^{\alpha\to\beta} (\lambda X.\mathbf{N}))$. Similarly, we have ensured $\vec{\nabla}_\mathfrak{f}$ holds when $* \in \{\beta\mathfrak{f}, \beta\mathfrak{f}\mathfrak{b}\}$ since $\neg(\mathbf{F}w_\alpha^\delta \doteq^\beta \mathbf{G}w_\alpha^\delta) \in \mathcal{H}$ whenever $\neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G}) \in \mathcal{H}$ where δ is the ordinal such that $\mathbf{A}^\delta \equiv \neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G})$.

It only remains to show that $\mathcal{H}$ is maximal in Γ_Σ. So, let $\mathbf{A} \in cwff_o(\Sigma)$ and $\mathcal{H} * \mathbf{A} \in \Gamma_\Sigma$ be given. Note that $\mathbf{A} \equiv \mathbf{A}^\delta$ for some $\delta < \epsilon$. Since $\mathcal{H}$ is closed under subsets we know that $\mathcal{H}^\delta * \mathbf{A}^\delta \in \Gamma_\Sigma$. By definition of $\mathcal{H}^{\delta+1}$ we conclude that $\mathbf{A}^\delta \in \mathcal{H}^{\delta+1}$ and hence $\mathbf{A} \in \mathcal{H}$.

So, Lemma 1.6.22 implies $\mathcal{H} \in \mathfrak{Hint}_*$ and $\mathcal{H}$ is saturated if Γ_Σ is saturated. $\Box$

We now use the Σ-Hintikka sets, guaranteed by Lemma 2.4.4, to construct a Σ-valuation for the Σ-term evaluation that turns it into a model.

Theorem 1.6.34 (Model Existence Theorem for Saturated Sets)
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* $\mathcal{H}$ *is a saturated Hintikka set in* $\mathfrak{Hint}_*$

(cf. Definition 1.6.21), then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ (cf. Definition 1.3.49) that satisfies $\mathcal{H}$. Furthermore, each domain $\mathcal{D}_\alpha$ of $\mathcal{M}$ has cardinality at most $\aleph_s$.

Proof: We start with the construction of a Σ-model $\mathcal{M}_1^{\mathcal{H}}$ for $\mathcal{H}$ based on the term evaluation $\mathcal{TE}(\Sigma)^\beta$. This model may not be in the model class $\mathfrak{M}_*$ as it may not satisfy property $\mathfrak{q}$. However, we will be able to use Theorem 1.3.62 to obtain a model of $\mathcal{H}$ which is.

Note that since $\mathcal{H}$ is saturated, by Lemma 1.6.26, $\mathcal{H}$ satisfies $\overline{\nabla_\neg}$, $\overline{\nabla_\vee}$, and $\overline{\nabla_\forall}$.

The domain of type α of the evaluation $\mathcal{TE}(\Sigma)^\beta$ (cf. Definition 1.3.35 and Lemma 1.3.36) is $cwff_\alpha(\Sigma){\downarrow_\beta}$, which has cardinality $\aleph_s$. To construct $\mathcal{M}_1^{\mathcal{H}}$, we simply need to give a valuation function for this evaluation. This valuation function should be a function $\upsilon\colon cwff_o(\Sigma){\downarrow_\beta} \longrightarrow \{\mathtt{T},\mathtt{F}\}$. We define

$$\upsilon(\mathbf{A}) := \begin{cases} \mathtt{T} & \text{if } \mathbf{A} \in \mathcal{H} \\ \mathtt{F} & \text{if } \mathbf{A} \notin \mathcal{H} \end{cases}$$

To show υ is a valuation, we must check the logical constants are interpreted appropriately. For each $\mathbf{A} \in cwff_o(\Sigma){\downarrow_\beta}$, we have $\upsilon(\neg\mathbf{A}) \equiv \mathtt{T}$ iff $\upsilon(\mathbf{A}) \equiv \mathtt{F}$ since $\neg\mathbf{A} \in \mathcal{H}$ iff $\mathbf{A} \notin \mathcal{H}$ by $\overline{\nabla_\neg}$. For each $\mathbf{A},\mathbf{B} \in cwff_o(\Sigma){\downarrow_\beta}$, we have $\upsilon(\mathbf{A} \vee \mathbf{B}) \equiv \mathtt{T}$ iff $\upsilon(\mathbf{A}) \equiv \mathtt{T}$ or $\upsilon(\mathbf{B}) \equiv \mathtt{T}$, since $(\mathbf{A} \vee \mathbf{B}) \in \mathcal{H}$ iff $\mathbf{A} \in \mathcal{H}$ or $\mathbf{B} \in \mathcal{H}$ by $\overline{\nabla_\vee}$. Finally, for each type α and $\mathbf{F} \in cwff_{\alpha\to o}(\Sigma){\downarrow_\beta}$, $\overline{\nabla_\forall}^\beta$ implies $(\Pi^\alpha\mathbf{F}) \in \mathcal{H}$ iff $(\mathbf{F}\mathbf{A}){\downarrow_\beta} \in \mathcal{H}$ for every $\mathbf{A} \in cwff_\alpha(\Sigma){\downarrow_\beta}$. Thus, we have $\upsilon(\Pi^\alpha\mathbf{F}) \equiv \mathtt{T}$ iff $\upsilon(\mathbf{F}@^\beta\mathbf{A}) \equiv \mathtt{T}$ for every $\mathbf{A} \in cwff_\alpha(\Sigma){\downarrow_\beta}$.

This verifies $\mathcal{M}_1^{\mathcal{H}} := (cwff(\Sigma){\downarrow_\beta}, @^\beta, \mathcal{E}^\beta, \upsilon)$ is a Σ-model. Clearly, $\mathcal{M}_1^{\mathcal{H}} \models \mathcal{H}$ since $\upsilon(\mathbf{A}) \equiv \mathtt{T}$ for every $\mathbf{A} \in \mathcal{H}$ by definition.

By Theorem 1.3.62, we have a congruence relation $\dot\sim$ on $\mathcal{M}_1^{\mathcal{H}}$ induced by Leibniz equality. Note that by Lemma 1.3.61 in the term model $\mathcal{M}_1^{\mathcal{H}}$, for every type α and every $\mathbf{A},\mathbf{B} \in cwff_\alpha(\Sigma){\downarrow_\beta}$, we have $\mathbf{A}_\alpha \dot\sim \mathbf{B}_\alpha$, iff $\upsilon(\mathbf{A} \doteq \mathbf{B}) \equiv \mathtt{T}$, iff $(\mathbf{A} \doteq^\alpha \mathbf{B}) \in \mathcal{H}$. Furthermore, if primitive equality is in the signature, then $\mathcal{H} \in \mathfrak{Hint}_*$ is a Hintikka set with primitive equality. Hence, $\mathcal{H}$ satisfies $\overline{\nabla_=^{\doteq}}$ by Lemma 1.6.23. We have $\mathbf{A} \dot\sim \mathbf{B}$, iff $(\mathbf{A} \doteq^\alpha \mathbf{B}) \in \mathcal{H}$, iff (by $\overline{\nabla_=^{\doteq}}$) $(\mathbf{A} =^\alpha \mathbf{B}) \in \mathcal{H}$, iff $\upsilon(\mathcal{E}(=^\alpha)@\mathbf{A}@\mathbf{B}) \equiv \mathtt{T}$.

Let $\mathcal{M} := \mathcal{M}_1^{\mathcal{H}}/_{\dot\sim}$. Each domain of this model has cardinality at most $\aleph_s$ as it is the quotient of a set of cardinality $\aleph_s$. By Theorem 1.3.62, we know the quotient model $\mathcal{M}$ will still model $\mathcal{H}$, satisfy property $\mathfrak{q}$, and be a model with primitive equality (if primitive equality is in the signature). Hence, $\mathcal{M} \in \mathfrak{M}_\beta$. Now, we can use Lemma 1.3.58 to check $\mathcal{M} \in \mathfrak{M}_*$ by checking certain properties of $\dot\sim$.

- $\mathfrak{b}$ When $* \in \{\beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, we must check that $\dot\sim$ has only two equivalence classes in $\mathcal{D}_o^\beta$. To show this, first note that $\overline{\nabla_\mathfrak{b}}$ holds for $\mathcal{H}$ by Lemma 1.6.27. Choose any β-normal $\mathbf{B} \in \mathcal{H}$. By $\vec{\nabla}_c$, $\neg\mathbf{B} \notin \mathcal{H}$. By $\overline{\nabla_\mathfrak{b}}$, for every $\mathbf{A} \in cwff_o(\Sigma){\downarrow_\beta}$ either $(\mathbf{A} \doteq^o \mathbf{B})$ or $(\mathbf{A} \doteq^o \neg\mathbf{B})$. That is, in $\mathcal{M}_1^{\mathcal{H}}$, for every $\mathbf{A} \in cwff_o(\Sigma){\downarrow_\beta}$ we either have $\mathbf{A} \dot\sim \mathbf{B}$ or $\mathbf{A} \dot\sim \neg\mathbf{B}$. So, we know $\mathcal{M}$ satisfies property $\mathfrak{b}$.
- η When $* \in \{\beta\eta, \beta\eta\mathfrak{b}\}$, the fact that $\dot\sim$ satisfies property η follows from $\overline{\nabla_\eta}$ which holds for $\mathcal{H}$ by Lemma 1.6.28.
- ξ Let $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$. We need to show that $\dot\sim$ satisfies property ξ. Let $\mathbf{M},\mathbf{N} \in wff_\beta(\Sigma)$, an assignment φ and a variable X_α be given. Suppose $\mathcal{E}_{\varphi,[\mathbf{A}/X]}(\mathbf{M}) \dot\sim \mathcal{E}_{\varphi,[\mathbf{A}/X]}(\mathbf{N})$ for every $\mathbf{A} \in cwff_\alpha(\Sigma){\downarrow_\beta}$. Let θ be the substitution defined by $\theta(Y) := \varphi(Y)$ for each variable

$Y \in (free(\mathbf{M}) \cup free(\mathbf{N})) \setminus \{X\}$. So, for each $\mathbf{A} \in cwff_\alpha(\Sigma)\big\downarrow_\beta$,

$$([\mathbf{A}/X]\theta(\mathbf{M}))\big\downarrow_\beta \equiv \mathcal{E}_{\varphi,[\mathbf{A}/X]}(\mathbf{M}) \mathrel{\dot\sim} \mathcal{E}_{\varphi,[\mathbf{A}/X]}(\mathbf{N}) \equiv ([\mathbf{A}/X]\theta(\mathbf{N}))\big\downarrow_\beta$$

That is, $([\mathbf{A}/X]\theta(\mathbf{M}) \doteq^\beta [\mathbf{A}/X]\theta(\mathbf{N}))\big\downarrow_\beta \in \mathcal{H}$ for every $\mathbf{A} \in cwff_\alpha(\Sigma)\big\downarrow_\beta$. By $\overline{\nabla}_\xi^\beta$ (Lemma 1.6.29), we have $((\lambda X.\theta(\mathbf{M})) \doteq^{\alpha\to\beta} \lambda X.\theta(\mathbf{N}))\big\downarrow_\beta \in \mathcal{H}$. So,

$$\mathcal{E}_\varphi(\lambda X.\mathbf{M}) \equiv (\lambda X.\theta(\mathbf{M}))\big\downarrow_\beta \mathrel{\dot\sim} (\lambda X.\theta(\mathbf{N}))\big\downarrow_\beta \equiv \mathcal{E}_\varphi(\lambda X.\mathbf{N}).$$

Thus, $\dot\sim$ satisfies ξ as desired.

$\mathfrak{f}$ When $* \in \{\beta\mathfrak{f}, \beta\mathfrak{fb}\}$, we must show $\dot\sim$ is functional. Let α and β be types and $\mathbf{G}, \mathbf{H} \in cwff_{\alpha\to\beta}(\Sigma)\big\downarrow_\beta$. We need to show $\mathbf{G} \mathrel{\dot\sim} \mathbf{H}$ iff $(\mathbf{GA})\big\downarrow_\beta \mathrel{\dot\sim} (\mathbf{HB})\big\downarrow_\beta$ for every $\mathbf{A}, \mathbf{B} \in cwff_\alpha(\Sigma)\big\downarrow_\beta$ with $\mathbf{A} \mathrel{\dot\sim} \mathbf{B}$. This follows directly from $\overline{\nabla}_{\mathfrak{f}}^\beta$.

This verifies the fact that $\mathcal{M} \in \mathfrak{M}_*$ whenever $\mathcal{H} \in \mathfrak{Hint}_*$. □

Theorem 1.6.35 (Model Existence Theorem)
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 $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$ we have: If Γ_Σ is an $\mathfrak{Acc}_*$ (cf. Definition 1.6.7), then there exists a model $\mathcal{M} \in \mathfrak{M}_*$ (cf. Definition 1.3.49) that satisfies Φ. Furthermore, each domain of $\mathcal{M}$ has cardinality at most $\aleph_s$.

Proof: Let Γ_Σ be an abstract consistency class. We can assume without loss of generality (cf. Lemma 1.6.19) that Γ_Σ is compact, so the preconditions of Lemma 2.4.4 are met. Therefore, there exists a saturated Hintikka set $\mathcal{H} \in \mathfrak{Hint}_*$ with $\Phi \subseteq \mathcal{H}$. The proof is completed by a simple appeal to the Theorem 1.6.34. □

Theorem 1.6.36 (Model Existence for Henkin Models)
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 (cf. Definition 1.3.50) that satisfies Φ. Furthermore, each domain of the model has cardinality at most $\aleph_s$.

Proof: By Theorem 1.6.35, there is a model $\mathcal{M} \in \mathfrak{M}_{\beta\mathfrak{fb}}$ with $\mathcal{M} \models \Phi$. By Theorem 1.3.69, there is a Henkin model $\mathcal{M}^{fr} \in \mathfrak{M}_{\beta\mathfrak{fb}}$ isomorphic to $\mathcal{M}$. By the isomorphism, we have $\mathcal{M}^{fr} \models \Phi$ and that each domain of $\mathcal{M}^{fr}$ has the same cardinality as the corresponding domain of $\mathcal{M}$. □

Remark 1.6.37 The model existence theorems show there are "enough" models in each class $\mathfrak{M}_*$ to model sufficiently pure sets in saturated abstract consistency classes in $\mathfrak{Acc}_*$. To complete the analysis, we should show there are not "too many" models. One way to show this is to define a class of sentences

$$\Gamma_\Sigma^* := \{\Phi | \exists \mathcal{M} \in \mathfrak{M}_*.\mathcal{M} \models \Phi\}$$

for each $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$ and show Γ_Σ^* is a (saturated) $\mathfrak{Acc}_*$. We only sketch the proof here.

The fact that each Γ_Σ^* satisfy ∇_c, ∇_β, $\nabla_\neg$, $\nabla_\vee$, $\nabla_\wedge$, $\nabla_\forall$, and ∇_{sat} is straightforward. The proof that $\nabla_\exists$ holds has the technical difficulty that one must modify the evaluation of a parameter. Showing $\nabla_\mathfrak{b}$ $[\nabla_\eta]$ holds when considering models with property $\mathfrak{b}$ $[\eta]$ is also easy.

When showing $\nabla_{\mathfrak{f}}$ holds in $\Gamma_\Sigma^{\beta\mathfrak{f}}$ [$\Gamma_\Sigma^{\beta\mathfrak{f}\mathfrak{b}}$], one sees the importance of assuming property $\mathfrak{q}$ holds. Suppose $\Phi \in \Gamma_\Sigma^{\beta\mathfrak{f}}$ [$\Gamma_\Sigma^{\beta\mathfrak{f}\mathfrak{b}}$] and $\neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G}) \in \Phi$. Then there is a model $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon) \in \mathfrak{M}_{\beta\mathfrak{f}}$ [$\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$] such that $\mathcal{M} \models \Phi$. This implies $\mathcal{M} \models \neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G})$. Without using property $\mathfrak{q}$, it follows by Lemma 1.4.2(1) that $\mathcal{E}(\mathbf{F}) \not\equiv \mathcal{E}(\mathbf{G})$. By functionality, there is an $\mathsf{a} \in \mathcal{D}_\alpha$ such that $\mathcal{E}(\mathbf{F})@\mathsf{a} \not\equiv \mathcal{E}(\mathbf{G})@\mathsf{a}$. Let φ be any assignment into $\mathcal{M}$. Then $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{F}X) \not\equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{G}X)$. Now, using property $\mathfrak{q}$, we can conclude $\mathcal{M}_{\varphi,[\mathsf{a}/X]} \models \neg((\mathbf{F}X) \doteq^{\beta} (\mathbf{G}X))$. Let $w_\alpha \in \Sigma$ be a parameter that does not occur in Φ. With some technical work which we omit, one can change the evaluation function to $\mathcal{E}'$ so that $\mathcal{E}'(\mathbf{A}) \equiv \mathcal{E}(\mathbf{A})$ for all $\mathbf{A} \in \Phi$, and $\mathcal{E}'(w) \equiv \mathsf{a}$. In the new model $\mathcal{M}' \equiv (\mathcal{D}, @, \mathcal{E}', \upsilon)$, we have $\mathcal{M}' \models \Phi$ and $\mathcal{M}' \models \neg(\mathbf{F}w \doteq^{\beta} \mathbf{G}w)$. Also, $\mathcal{M}' \in \mathfrak{Acc}_{\beta\mathfrak{f}}$ [$\mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$]. This shows $\Phi * \neg(\mathbf{F}w \doteq^{\beta} \mathbf{G}w) \in \Gamma_\Sigma^{\beta\mathfrak{f}}$ [$\Gamma_\Sigma^{\beta\mathfrak{f}\mathfrak{b}}$]. The proof that ∇_ξ holds in $\Gamma_\Sigma^{\beta\xi}$ [$\Gamma_\Sigma^{\beta\xi\mathfrak{b}}$] is analogous.

We have now established a set of proof-theoretic conditions that are sufficient to guarantee the existence of a model.

1.7 Characterizing Higher Order Natural Deduction Calculi

In this section we apply the model existence theorems above to prove some classical higher-order calculi of natural deduction sound and complete with respect to the model classes introduced in Section 1.3. The first calculus for such a formulation of higher order logic was a Hilbert-style system introduced by Alonzo Church in [Chu40][8]. Leon Henkin proves completeness (with respect to Henkin-models) for a similar calculus with full extensionality in [Hen50]. Peter Andrews introduced a weaker calculus $\mathfrak{T}_\beta$ [And71], which lacks all forms of extensionality. This calculus has been widely used as a syntactic measure of completeness for machine-oriented calculi [And71, Hue72, Hue73, JP72, Mil83, Koh94, Koh95].

Instead of applying our methods to Hilbert-style calculi, we will use a collection of natural deduction calculi to avoid the tedious details of proving a deduction theorem and propositional completeness. Moreover, natural deduction calculi are more relevant in practice, they form the logical basis for semi-automated theorem proving systems such as HOL [GM93], Isabelle [NPW02], or Ωmega [Sea02].

Definition 1.7.1 (The Calculi $\mathfrak{NK}_*$) The calculus $\mathfrak{NK}_\beta$ consists of the inference rules in Figure 1.6 for the provability judgment $\Vdash$ between sets of sentences Φ and sentences $\mathbf{A}$. (We write $\Vdash \mathbf{A}$ for $\emptyset \Vdash \mathbf{A}$.) The rule $\mathfrak{NK}(\beta)$ incorporates β-equality into $\Vdash$. The others characterize the semantics of the connectives and quantifiers.

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 rules[9] specified in Figure 1.7 when specified in $*$.

Note that $\mathfrak{NK}_\beta$ and $\mathfrak{NK}_{\beta\mathfrak{f}\mathfrak{b}}$ correspond to the extremes of the model classes discussed in Section 1.3 (cf. Figure 1.1 in the introduction). Standard models do not admit (recursively axiomatizable) calculi that are sound and complete, $\mathfrak{NK}_{\beta\mathfrak{f}\mathfrak{b}}$ is complete for Henkin models, and $\mathfrak{NK}_\beta$ is complete for $\mathfrak{M}_\beta$. We will now show soundness and completeness of each $\mathfrak{NK}_*$ with respect to each corresponding model class $\mathfrak{M}_*$ by using the model existence theorems in Section 1.6.

[8] Church included functional extensionality axioms but only mentions the Boolean extensionality axiom as an option.

[9] Recall that $\mathbf{F}_o$ is defined to be $\neg(\forall P_o.(P \vee \neg P))$ and $\mathcal{M} \not\models \mathbf{F}_o$ for each Σ-model $\mathcal{M}$ (cf. Lemma 1.3.43).

$$\frac{\mathbf{A} \in \Phi}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\mathit{Hyp}) \qquad \frac{\mathbf{A} \equiv_{\beta} \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\beta)$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \neg\mathbf{A}}\,\mathfrak{NK}(\neg I) \qquad \frac{\Phi \Vdash \neg\mathbf{A} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\neg E)$$

$$\frac{\Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \vee \mathbf{B}}\,\mathfrak{NK}(\vee I_L) \qquad \frac{\Phi \Vdash \mathbf{B}}{\Phi \Vdash \mathbf{A} \vee \mathbf{B}}\,\mathfrak{NK}(\vee I_R)$$

$$\frac{\Phi \Vdash \mathbf{A} \vee \mathbf{B} \quad \Phi * \mathbf{A} \Vdash \mathbf{C} \quad \Phi * \mathbf{B} \Vdash \mathbf{C}}{\Phi \Vdash \mathbf{C}}\,\mathfrak{NK}(\vee E)$$

$$\frac{\Phi \Vdash \mathbf{G} w_\alpha \quad w \text{ parameter not occurring in } \Phi \text{ or } \mathbf{G}}{\Phi \Vdash \Pi^\alpha \mathbf{G}}\,\mathfrak{NK}(\mathit{\Pi I})^w$$

$$\frac{\Phi \Vdash \Pi^\alpha \mathbf{G}}{\Phi \Vdash \mathbf{G}\mathbf{A}}\,\mathfrak{NK}(\mathit{\Pi E}) \qquad \frac{\Phi * \neg\mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\mathit{Contr})$$

Figure 1.6: Inference rules for $\mathfrak{NK}_\beta$

$$\frac{\mathbf{A} \equiv_{\beta\eta} \mathbf{B} \quad \Phi \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{B}}\,\mathfrak{NK}(\eta) \qquad \frac{\Phi \Vdash \forall X_\alpha . \mathbf{M} \doteq^{\beta} \mathbf{N}}{\Phi \Vdash (\lambda X_\alpha . \mathbf{M}) \doteq^{\alpha\to\beta} (\lambda X_\alpha . \mathbf{N})}\,\mathfrak{NK}(\xi)$$

$$\frac{\Phi \Vdash \forall X_\alpha . \mathbf{G} X \doteq^{\beta} \mathbf{H} X}{\Phi \Vdash \mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}}\,\mathfrak{NK}(\mathfrak{f})$$

$$\frac{\Phi * \mathbf{A} \Vdash \mathbf{B} \quad \Phi * \mathbf{B} \Vdash \mathbf{A}}{\Phi \Vdash \mathbf{A} \doteq^{o} \mathbf{B}}\,\mathfrak{NK}(\mathfrak{b}) \qquad \frac{\Phi * \neg\mathbf{A} \Vdash \mathbf{F}_o}{\Phi \Vdash \mathbf{A}}\,\mathfrak{NK}(\mathit{Contr})$$

Figure 1.7: Inference rules for $\mathfrak{NK}_\beta$

Theorem 1.7.2 (Soundness)
$\mathfrak{NK}_*$ is sound for $\mathfrak{M}_*$ 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 $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ in $\mathfrak{M}_*$ such that $\mathcal{M} \models \Phi$.

Proof: This can be shown by a simple induction on the derivation of $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{C}$. We distinguish based on the last rule of the derivation. The only base case is $\mathfrak{NK}(Hyp)$, which is trivial since $\mathcal{M} \models \mathbf{C}$. whenever $\mathcal{M} \models \Phi$ and $\mathbf{C} \in \Phi$.

$\mathfrak{NK}(\beta)$: Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \mathbf{A}$ and $\mathbf{A} {\equiv_\beta} \mathbf{C}$. Let $\mathcal{M}$ be a model of Φ. By induction, we know $\mathcal{M} \models \mathbf{A}$ and so $\mathcal{M} \models \mathbf{C}$ using Remark 1.3.18.

$\mathfrak{NK}(Contr)$: Suppose $\mathcal{M} \models \Phi$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi * \neg\mathbf{C} \Vdash \mathbf{F}_o$. By Lemma 1.3.43, $\mathcal{M} \not\models \mathbf{F}_o$. So, we must have $\mathcal{M} \not\models \neg\mathbf{C}$. Hence, $\mathcal{M} \models \mathbf{C}$.

$\mathfrak{NK}(\neg I)$: Analogous to $\mathfrak{NK}(Contr)$.

$\mathfrak{NK}(\neg E)$: Suppose $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \neg\mathbf{A}$ and $\Phi \Vdash \mathbf{A}$. By induction, any model of Φ would have to model both $\mathbf{A}$ and $\neg\mathbf{A}$. So, there is no model of Φ and we are done.

$\mathfrak{NK}(\vee I_L)$: Suppose $\mathcal{M} \models \Phi$, $\mathbf{C}$ is $(\mathbf{A} \vee \mathbf{B})$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \mathbf{A}$. By induction, $\mathcal{M} \models \mathbf{A}$ and so $\mathcal{M} \models (\mathbf{A} \vee \mathbf{B})$.

$\mathfrak{NK}(\vee I_R)$: Analogous to $\mathfrak{NK}(\vee I_L)$.

$\mathfrak{NK}(\vee E)$: 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 $\mathcal{M}$ 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, $\Phi \Vdash \mathbf{C}$.

$\mathfrak{NK}(\Pi I)$: Suppose $\mathbf{C}$ is $(\Pi^\alpha \mathbf{G})$ and $\Phi \Vdash (\Pi^\alpha \mathbf{G})$ follows from $\Phi \Vdash \mathbf{G}w$ where w_α is a parameter which does not occur in any sentence in Φ or in $\mathbf{G}$. Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a model of Φ. Assume $\mathcal{M} \not\models \Pi^\alpha \mathbf{G}$. Then there must exist some $\mathsf{a} \in \mathcal{D}_\alpha$ such that $\upsilon(\mathcal{E}(\mathbf{G})@\mathsf{a}) \equiv \mathtt{F}$. From the evaluation function $\mathcal{E}$, one can define another evaluation function $\mathcal{E}'$ such that $\mathcal{E}'(w) \equiv \mathsf{a}$ and $\mathcal{E}'_\varphi(\mathbf{A}_\alpha) \equiv \mathcal{E}_\varphi(\mathbf{A}_\alpha)$ if w does not occur in $\mathbf{A}$. Let $\mathcal{M}' := (\mathcal{D}, @, \mathcal{E}', \upsilon)$. Since $\mathcal{M}' \models \Phi$, by induction we have $\mathcal{M}' \models \mathbf{G}w$. This contradicts $\upsilon(\mathcal{E}'(\mathbf{G})@\mathsf{a}) \equiv \upsilon(\mathcal{E}(\mathbf{G})@\mathsf{a}) \equiv \mathtt{F}$. Thus, $\mathcal{M} \models \Pi^\alpha \mathbf{G}$.

$\mathfrak{NK}(\Pi E)$: Suppose $\mathbf{C}$ is $(\mathbf{GA})$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash (\Pi^\alpha \mathbf{G})$. Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a model of Φ. By induction, $\mathcal{M} \models (\Pi^\alpha \mathbf{G})$ and thus $\upsilon(\mathcal{E}(\mathbf{G}))@\mathsf{a} \equiv \mathtt{T}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$. In particular, $\mathcal{M} \models \mathbf{GA}$.

We now check soundness of the rules in Figure 1.7 with respect to their model classes:

$\mathfrak{NK}(\eta)$: Analogous to $\mathfrak{NK}(\beta)$ using property η.

$\mathfrak{NK}(\xi)$: Suppose $\mathbf{C}$ is $(\lambda X_\alpha . \mathbf{M}) \doteq^{\alpha\to\beta} (\lambda X_\alpha . \mathbf{N})$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \forall X_\alpha . \mathbf{M} \doteq^\beta \mathbf{N}$. Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a model of Φ. By induction, we have $\mathcal{M} \models \forall X_\alpha . \mathbf{M} \doteq^\beta \mathbf{N}$. So, for any assignment φ and $\mathsf{a} \in \mathcal{D}_\alpha$, $\mathcal{M} \models_{\varphi,[\mathsf{a}/X]} \mathbf{M} \doteq^\beta \mathbf{N}$. By Lemma 1.4.2(2), $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N})$. By property ξ, $\mathcal{E}_\varphi(\lambda X_\alpha . \mathbf{M}) \equiv \mathcal{E}_\varphi(\lambda X_\alpha . \mathbf{N})$ and thus $\mathcal{M} \models \mathbf{C}$ by Lemma 1.4.2(1).

$\mathfrak{NK}(\mathfrak{f})$: Suppose $\mathbf{C}$ is $\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H}$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi \Vdash \forall X_\alpha . \mathbf{G}X \doteq^\beta \mathbf{H}X$. Let $\mathcal{M}$ be a model of Φ. By induction, we know $\mathcal{M} \models \forall X_\alpha . \mathbf{G}X \doteq^\beta \mathbf{H}X$. By Theorem 1.4.3(3), we must have $\mathcal{M} \models (\mathbf{G} \doteq^{\alpha\to\beta} \mathbf{H})$.

$\mathfrak{NK}(\mathfrak{b})$: Suppose $\mathbf{C}$ is $\mathbf{A} \doteq^o \mathbf{B}$ and $\Phi \Vdash \mathbf{C}$ follows from $\Phi * \mathbf{A} \Vdash \mathbf{B}$ and $\Phi * \mathbf{B} \Vdash \mathbf{A}$. Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ 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 $\upsilon(\mathcal{E}(\mathbf{A})) \equiv \upsilon(\mathcal{E}(\mathbf{B}))$. By Lemma 1.3.48, we have $\mathcal{M} \models (\mathbf{A} \Leftrightarrow \mathbf{B})$. By Theorem 1.4.3(4), we must have $\mathcal{M} \models (\mathbf{A} \doteq^o \mathbf{B})$. $\square$

Definition 1.7.3 ($\mathfrak{NK}_*$-Consistent) A set of sentences Φ is $\mathfrak{NK}_*$-*inconsistent* if $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{F}_o$, and $\mathfrak{NK}_*$-*consistent* otherwise.

Now, we use the model existence theorems for $\mathcal{HOL}$ to give short and elegant proofs of completeness for $\mathfrak{NK}_*$.

Lemma 1.7.4 *The class* $\Gamma^*_\Sigma := \{\Phi \subset cwff_o(\Sigma) \mid \Phi \text{ is } \mathfrak{NK}_*\text{-consistent}\}$ *is a saturated* $\mathfrak{Acc}_*$.

Proof: Obviously Γ^*_Σ is closed under subsets, since any subset of a $\mathfrak{NK}_*$-consistent set is $\mathfrak{NK}_*$-consistent. We now check the remaining conditions. We prove all the properties (except $\nabla_\mathfrak{b}$) by proving the contrapositive.

∇_c: Suppose $\mathbf{A}, \neg\mathbf{A} \in \Phi$. We have $\Phi \Vdash \mathbf{F}_o$ by $\mathfrak{NK}(Hyp)$ and $\mathfrak{NK}(\neg E)$.

∇_β: Let $\mathbf{A} \in \Phi$ and $\Phi * \mathbf{A}{\downarrow_\beta}$ be $\mathfrak{NK}_*$-inconsistent. That is, $\Phi * \mathbf{A}{\downarrow_\beta} \Vdash \mathbf{F}_o$. By $\mathfrak{NK}(\neg I)$, we know $\Phi \Vdash \neg\, \mathbf{A}{\downarrow_\beta}$. Since $\mathbf{A} \in \Phi$, we know $\Phi \Vdash \mathbf{A}{\downarrow_\beta}$ by $\mathfrak{NK}(Hyp)$ and $\mathfrak{NK}(\beta)$. So, by $\mathfrak{NK}(\neg E)$ we know $\Phi \Vdash \mathbf{F}_o$ and Φ is $\mathfrak{NK}_*$-inconsistent.

$\nabla_\neg$: Suppose $\neg\neg\mathbf{A} \in \Phi$ and $\Phi * \mathbf{A}$ is $\mathfrak{NK}_*$-inconsistent. From $\Phi * \mathbf{A} \Vdash \mathbf{F}_o$ and $\mathfrak{NK}(\neg I)$, we have $\Phi \Vdash \neg\mathbf{A}$. Since $\neg\neg\mathbf{A} \in \Phi$, we can apply $\mathfrak{NK}(Hyp)$ and $\mathfrak{NK}(\neg E)$ to obtain $\Phi \Vdash \mathbf{F}_o$.

$\nabla_\vee$: Suppose $(\mathbf{A} \vee \mathbf{B}) \in \Phi$ and both $\Phi * \mathbf{A}$ and $\Phi * \mathbf{B}$ are $\mathfrak{NK}_*$-inconsistent. By $\mathfrak{NK}(Hyp)$ and $\mathfrak{NK}(\vee E)$, we have $\Phi \Vdash \mathbf{F}_o$.

$\nabla_\wedge$: Suppose $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$ and $\Phi * \neg\mathbf{A}$ is $\mathfrak{NK}_*$-inconsistent. By $\mathfrak{NK}(Contr)$ and $\mathfrak{NK}(\vee I_L)$, we have $\Phi \Vdash \mathbf{A} \vee \mathbf{B}$. Using $\mathfrak{NK}(\neg E)$ with $\neg(\mathbf{A} \vee \mathbf{B}) \in \Phi$, Φ is $\mathfrak{NK}_*$-inconsistent. Similarly, if $\Phi * \neg\mathbf{B}$ is $\mathfrak{NK}_*$-inconsistent, then so is Φ.

$\nabla_\forall$: Suppose $(\Pi^\alpha \mathbf{G}) \in \Phi$ and $\Phi * (\mathbf{GA})$ is $\mathfrak{NK}_*$-inconsistent. By $\mathfrak{NK}(\neg E)$, $\Phi \Vdash \neg(\mathbf{GA})$. By $\mathfrak{NK}(Hyp)$ and $\mathfrak{NK}(\Pi E)$, $\Phi \Vdash \mathbf{GA}$. Finally, $\mathfrak{NK}(\neg E)$ implies $\Phi \Vdash \mathbf{F}_o$.

$\nabla_\exists$: 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. By $\mathfrak{NK}(\neg I)$, $\Phi \Vdash \mathbf{G}w$. By $\mathfrak{NK}(\Pi I)$, $\Phi \Vdash (\Pi^\alpha \mathbf{G})$. Using $\mathfrak{NK}(\neg E)$ with $\neg(\Pi^\alpha \mathbf{G}) \in \Phi$, Φ is $\mathfrak{NK}_*$-inconsistent.

∇_{sat}: Let $\Phi * \mathbf{A}$ and $\Phi * \neg\mathbf{A}$ be $\mathfrak{NK}_*$-inconsistent. We show that Φ is $\mathfrak{NK}_*$-inconsistent. Using $\mathfrak{NK}(\neg I)$, we know $\Phi \Vdash \neg\mathbf{A}$ and $\Phi \Vdash \neg\neg\mathbf{A}$. By $\mathfrak{NK}(\neg E)$, we have $\Phi \Vdash \mathbf{F}_o$.

Thus we have shown that Γ^β_Σ is saturated and in $\mathfrak{Acc}_\beta$. Now let us check the conditions for the additional properties η, ξ, $\mathfrak{f}$, and $\mathfrak{b}$.

∇_η: If $*$ includes η, then the proof proceeds as in ∇_β above, but with the rule $\mathfrak{NK}(\eta)$.

∇_ξ: 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 Φ. By $\mathfrak{NK}(Contr)$, we have $\Phi \Vdash ([w/X]\mathbf{M} \doteq^\beta [w/X]\mathbf{N})$. By $\mathfrak{NK}(\Pi I)$ and $\mathfrak{NK}(\xi)$, $\Phi \Vdash (\lambda X.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X.\mathbf{N})$. By $\mathfrak{NK}(\neg E)$, Φ is $\mathfrak{NK}_*$-inconsistent.

$\nabla_{\mathfrak{f}}$: This case is analogous to the previous one, generalizing $\lambda X.\mathbf{M} \doteq \lambda X.\mathbf{N}$ to arbitrary $\mathbf{G} \doteq \mathbf{H}$ and using the extensionality rule $\mathfrak{NK}(\mathfrak{f})$ instead of $\mathfrak{NK}(\xi)$.

$\nabla_{\mathfrak{b}}$: Suppose $*$ includes $\mathfrak{b}$. We argue by contradiction. Assume that $\neg\mathbf{A} \doteq^o \mathbf{B} \in \Phi$ but both $\Phi * \neg\mathbf{A} * \mathbf{B} \notin \Gamma^*_\Sigma$ and $\Phi * \mathbf{A} * \neg\mathbf{B} \notin \Gamma^*_\Sigma$. So both are $\mathfrak{NK}_*$-inconsistent and we have $\Phi * \mathbf{A} \Vdash \mathbf{B}$ and $\Phi * \mathbf{B} \Vdash \mathbf{A}$ by $\mathfrak{NK}(\mathit{Contr})$. By $\mathfrak{NK}(\mathfrak{b})$, we have $\Phi \Vdash (\mathbf{A} \doteq^o \mathbf{B})$. Since $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Phi$, Φ is $\mathfrak{NK}_*$-inconsistent. □

Theorem 1.7.5 (Henkin's Theorem for $\mathfrak{NK}_*$)
Let $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, then every sufficiently Σ-pure $\mathfrak{NK}_*$-consistent set of sentences has an $\mathfrak{M}_*$-model.

Proof: Let Φ be a sufficiently Σ-pure $\mathfrak{NK}_*$-consistent set of sentences. By Theorem 1.7.4 we know that the class of sets of $\mathfrak{NK}_*$-consistent sentences constitute a saturated $\mathfrak{Acc}_*$, thus the Model Existence Theorem (Theorem 1.6.35) guarantees a $\mathfrak{M}_*$ model for Φ. □

Corollary 1.7.6 (Completeness Theorem for $\mathfrak{NK}_*$)
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 $\mathcal{M} \in \mathfrak{M}_*$ that satisfy Φ, then $\Phi \Vdash_{\mathfrak{NK}_*} \mathbf{A}$.

Proof: Let $\mathbf{A}$ be given such that $\mathbf{A}$ is valid in all $\mathfrak{M}_*$ models that satisfy Φ. So, $\Phi * \neg\mathbf{A}$ is unsatisfiable in $\mathfrak{M}_*$. 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 $\mathfrak{NK}(\mathit{Contr})$. □

Finally we can use the completeness theorems obtained so far to prove a compactness theorem for our semantics.

Corollary 1.7.7 (Compactness Theorem for $\mathfrak{NK}_*$)
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 $\mathfrak{M}_*$-model iff every finite subset of Φ has an $\mathfrak{M}_*$-model.

Proof: If Φ has no $\mathfrak{M}_*$-model, then by Corollary 1.7.6 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 $\mathfrak{M}_*$-model. □

Remark 1.7.8 (Calculi with Primitive Equality) If primitive equality is included in the signature, a simple way of extending the calculi $\mathfrak{NK}_*$ in a sound and complete way is to include the rules $\mathfrak{NK}(=^r)$ and $\mathfrak{NK}(=^l)$ in Figure 1.8. These rules are clearly sound for models with primitive equality. One can argue completeness by showing $\Gamma^*_\Sigma := \{\Phi \subset \mathit{wff}_o(\Sigma) \mid \Phi \text{ is } \mathfrak{NK}_*\text{-consistent}\}$ is a saturated $\mathfrak{Acc}_*$ with primitive equality. By Lemma 1.7.4, we already know Γ^*_Σ is a saturated $\mathfrak{Acc}_*$. To show the conditions for primitive equality, one can show Γ^*_Σ satisfies $\nabla^r_=$ using $\mathfrak{NK}(=^r)$ and $\nabla^{\doteq}_=$ using $\mathfrak{NK}(=^l)$.

$$\frac{}{\Phi \Vdash \mathbf{A} =^{\alpha} \mathbf{A}}\,\mathfrak{NK}(=^{r}) \qquad \frac{\Phi \Vdash \mathbf{C} =^{\alpha} \mathbf{D}}{\Phi \Vdash \mathbf{C} \doteq^{\alpha} \mathbf{D}}\,\mathfrak{NK}(=^{l})$$

Figure 1.8: Primitive Equality in $\mathfrak{NK}_*$

1.8 Conclusion

In this article, we have given an overview of the landscape of semantics for classical higher order logics. We have differentiated nine different possible notions and have tied the discerning properties to conditions of corresponding abstract consistency classes. The practical relevance of these notions has been illustrated by pointing to application scenarios within mathematics, programming languages, and computational linguistics.

Our model existence theorems are strong proof tools connecting syntax and semantics. A standard application is in completeness analysis of higher order calculi. A calculus $\mathcal{C}$ is shown to be complete for a model class $\mathfrak{M}_*$ by showing that the class of $\mathcal{C}$-consistent or $\mathcal{C}$-irrefutable sets of sentences is in $\mathfrak{Acc}_*$. Then completeness follows from the model existence results. We have given an example of this by showing completeness for natural deduction calculi in Section 1.7.

1.8.1 Applications and Related Work

The generalized model classes $\mathfrak{M}_*$ have many possible applications. An example is higher order logic programming [NM94] where the denotational semantics of programs can induce non-standard meanings for the classical connectives. For instance, given a SLD-like search strategy as in λ-PROLOG [Mil91], conjunction is not commutative any more. Therefore, various authors have proposed model-theoretic semantics where property $\mathfrak{b}$ fails. David Wolfram, for instance, uses Andrews' υ-complexes [Wol94] as a semantics for λ-PROLOG and Gopalan Nadathur uses "labeled structures" for the same purpose in [NM94]. Mary DeMarco [DeM99] also develops a model theory for intuitionistic type theory and λ-prolog in which property $\mathfrak{b}$ may fail (James Lipton and Mary DeMarco are continuing this work). Till Mossakowski and Lutz Schröder have been studying non-functional Henkin models for a partial λ-calculus in the context of the HasCasl specification language [SM02, Sch02]. It is plausible to assume that the results of this article will be useful for further development in this direction. Further relevance of model-theoretic semantics where property $\mathfrak{q}$ fails, however, is not sufficiently investigated yet, but seems a promising line of research.

The article also provides a basis for the investigation of hyperintensional semantics of natural languages. In fact early versions of this article have already influenced [LP02]. Hyperintensional semantics provide theories for logics where Boolean extensionality (and thus the substitutability of equivalents) can fail. Linguistically motivated theories like the ones presented in [Tom80, CT88, LS95, LP02] introduce intensional (non-standard) variants of the connectives and quantifiers acting on a generalized domain of truth values. Interestingly, only [LS95] and [LP02] present formal model-theoretic semantics. The model construction in [LS95] strongly resembles Peter Andrew's υ-complexes (semantic objects are paired with syntactic representations; in this case linguistic parse trees). In [LP02], $\mathcal{D}_o$ is taken to be a pre-Boolean algebra, and possible worlds are associated with ultrafilters. A direct comparison is aggravated by the fact that Lappin and Pollard's work is situated in a Montague-style intensional (i.e. modal) context. A generalization of our work by techniques from [Fit02] seems the way to go here.

1.8.2 Relaxing the Saturation Assumption

Unfortunately, the model existence theorems presented in this article do not support completeness proofs for most higher order machine-oriented calculi, such as higher order resolution [Hue73, BK98], higher order paramodulation [Ben99b], or tableau-based calculi [And89, Koh95]. This is because we had to assume saturation of abstract consistency classes to prove the model existence theorems. The problem is that machine oriented calculi are typically, in some sense, cut-free. This makes saturation very difficult to show.

For the same reason the results of this article also do not apply to another prominent application of model existence theorems: relatively simple (but non-constructive) cut-elimination theorems. In [And71] Peter Andrews applies his "Unifying Principle" to cut-elimination in a cut-free non-extensional sequent calculus, by proving the calculus complete (relative to $\mathfrak{T}_\beta$). He concludes that cut-elimination is valid for this calculus. Again, the saturation condition prevents us from obtaining variants of the extensional cut-elimination theorems in [Tak68, Tak87] by Andrews' approach using our model existence theorem for Henkin models. In fact one can prove (cf. [BBK02b]) that the problem of showing that an abstract consistency class can be extended to a saturated one is equivalent to showing cut elimination for certain sequent or resolution calculi.

To account for the saturation problem we have additionally investigated model existence for the model classes presented in this article using an extension of Peter Andrews' υ-complexes (cf. [BBK02b]). The model construction in this technique requires an abstract consistency class to satisfy certain *acceptability* conditions which are much weaker than saturation. (For example, the acceptability conditions can be shown to hold for abstract consistency classes obtained from a certain cut-free sequent calculi.) Because this technique is much more complex and subtle than the relatively simple quotients of term evaluations used in this article, we did not include the extended results here. The unsaturated model existence theorems imply that every acceptable abstract consistency class can be extended to a saturated one. Armed with this fact, we can use the model existence theorems presented here to rescue the general completeness and cut elimination results mentioned above. To show, for example, completeness of a higher order machine-oriented calculus $\mathcal{C}$, we define the class Γ of $\mathcal{C}$-irrefutable sentences and show that it is an acceptable (but unsaturated) abstract consistency class. By the extension result in [BBK02b] there is a saturated abstract consistency class $\Gamma' \supseteq \Gamma$. By application of saturated model existence from this article we obtain a suitable model for every (sufficiently Σ-pure) $\Phi \in \Gamma'$ and thus for every (sufficiently Σ-pure) $\Phi \in \Gamma$. This immediately gives us completeness. Hence, the leverage added by this article together with [BBK02b] is that we can now extend non-extensional cut-elimination results to extensional cases.

Chapter 2

Semantic Techniques for Cut Elimination in Higher Order Logics

In this chapter , we extend the methods presented in the previous chapter and obtain analogous model existence results without assuming saturation. We will show that saturation can be as hard to prove as cut elimination. Consequently, the framework presented in the previous chapter is not easily applicable, for instance, to analyze the deductive power of machine oriented calculi. This limitation of the saturated abstract consistency approach is lifted by the extensions presented in this chapter.

2.1 Motivation

In chapter 1 we have given an overview of the landscape of semantics for classical higher order logics. We have differentiated several different model classes and have tied their discerning properties to conditions of corresponding abstract consistency classes.

The standard application for the model existence theorems is in the completeness analysis of higher order calculi. A calculus $\mathcal{C}$ is shown to be complete for a model class $\mathfrak{M}_\mathcal{C}$ by showing that the class of $\mathcal{C}$-consistent or $\mathcal{C}$-irrefutable sets of sentences is in $\mathfrak{Acc}_\mathcal{C}$. chapter 1 demonstrates this by showing completeness for a collection of natural deduction calculi that correspond to the developed landscape of model classes. Unfortunately, the model existence theorems presented in chapter 1 do not support completeness proofs for most higher order machine-oriented calculi, such as higher order resolution [Hue73, BK98] and tableau-based calculi [And89, Koh95], since we had to assume saturation of abstract consistency classes to prove them. The problem is that such calculi only allow one to show "atomic saturation" (a variant, where the saturation condition is only required for atomic formulae). Even though – at first glance – this seems to be a minor generalization of the results presented in chapter 1, this is not so. We will show that full saturation corresponds to admissibility of cut (which is hard to show for higher order calculi) while atomic saturation corresponds to admissibility of atomic cuts (which is not).

Another common application of model existence theorems is that they allow for relatively simple (but non-constructive) proofs of cut-elimination theorems. In [And71] Peter Andrews applies his "Unifying Principle" to cut-elimination in a non-extensional sequent calculus: He proves the calculus complete (relative to a Hilbert style calculus $\mathfrak{T}_\beta$) without the cut rule and concludes that cut-elimination is valid for this calculus. Again, the saturation condition prevents us from obtaining variants of the extensional cut-elimination theorems in [Tak68, Tak87], by Andrews' approach, using the model existence theorem of chapter 1. As we will show the problem of proving that an abstract consistency class can be extended to a saturated one is equivalent to proving cut elimination for certain sequent or resolution calculi. BegNP(1)

In this paper we present strong model existence theorems for the model classes presented in chapter 1 using an extension of Peter Andrews' υ-complexes [And71]. For this we weaken the saturation assumption from chapter 1 to certain natural "acceptability" conditions for abstract consistency classes – which are easy to establish for the abstract consistency classes that arise in completeness proofs for the calculi we have looked at. In addition, we show that every acceptable EndNP(1)
abstract consistency class can be extended to a saturated one. Armed with this *saturated extension theorem* (see Theorem 2.5.3) we can use the model existence theorems from chapter 1 to rescue the general completeness and cut elimination results mentioned above. To show e.g. completeness of a higher order machine-oriented calculus $\mathcal{C}$, we study the class Γ of $\mathcal{C}$-irrefutable sentences, and show that it is an acceptable abstract consistency class. By the saturated extension theorem there is a saturated abstract consistency class $\Gamma' \supseteq \Gamma$. Hence, by saturated model existence, there is a suitable model for every $\Phi \in \Gamma'$ and thus for every $\Phi \in \Gamma$; this immediately gives us completeness.

Let us now look at two examples that motivate the acceptibility conditions. These examples show that not every abstract consistency class can be extended to a saturated one.

Example 2.1.1 (Unsaturated $\mathfrak{Acc}_{\beta\mathfrak{b}}$ without saturated Extension)
Let Σ be a typed signature with $a, b, q \in \Sigma$ and $\Gamma_\Sigma := \{\Phi\}$, where $\Phi = \{a, b, (qa), \neg(qb)\}$. It is easy to check that this is an $\mathfrak{Acc}_{\beta\mathfrak{fb}}$, hence an $\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$ and $\mathfrak{Acc}_{\beta\mathfrak{b}}$. Suppose we have a saturated extension Γ'_Σ of Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{b}}$ [$\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$,$\mathfrak{Acc}_{\beta\mathfrak{fb}}$] and $\mathcal{H}$ is a Hintikka set in Γ'_Σ with $\Phi \subseteq \mathcal{H}$. Then we must either have $(a \doteq^o b) \in \mathcal{H}$ or $\neg(a \doteq^o b) \in \mathcal{H}$ by saturation. In the first case, applying $\nabla_\forall$ with the constant q,

[1] New Part: reformulated to better capture what we are doing

$\nabla_\vee$ and ∇_c contradicts $(qa), \neg(qb) \in \Phi \subseteq \mathcal{H}$. In the second case, $\nabla_\mathfrak{b}$ and ∇_c contradict $a, b \in \Phi \subseteq \mathcal{H}$. One can also extend this example by letting $\Phi := \{\neg(q_{o\to o} b_o)\} \cup \{\mathbf{A} | \mathbf{A} \text{ atomic and } \mathbf{A} \not\equiv (qb)\}$ to obtain an atomically saturated $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ which has no saturated extension.

Example 2.1.2 (Unsaturated $\mathfrak{Acc}_{\beta\mathfrak{f}}$ without saturated Extension)
Similar to the previous example we assume that $q_{(\iota\to\iota)\to o}, g_{\iota\to\iota}, h_{\iota\to\iota} \in \Sigma$ and choose and $\Gamma_\Sigma := \{\Phi\}$ where $\Phi = \{\neg qg\} \cup \{\mathbf{A} | \mathbf{A} \text{ atomic and } \mathbf{A} \not\equiv qg\}$. Again it is easy to check that this is an atomically saturated $\mathfrak{Acc}_{\beta\mathfrak{fb}}$, hence an $\mathfrak{Acc}_{\beta\mathfrak{f}}$. Suppose we have a saturated extension Γ'_Σ of Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{f}}$ [$\mathfrak{Acc}_{\beta\mathfrak{fb}}$] and $\mathcal{H}$ is a Hintikka set in Γ'_Σ with $\Phi \subseteq \mathcal{H}$. Then we must either have $(g \doteq^{\iota\to\iota} h) \in \mathcal{H}$ or $\neg(g \doteq^{\iota\to\iota} h) \in \mathcal{H}$ by saturation. In the first case, we obtain a contradiction to $\neg(qg), (qh) \in \Phi \subseteq \mathcal{H}$ using $\nabla_\forall$ with constant q, $\nabla_\vee$ and ∇_c. In the second case, $\neg(g \doteq^{\iota\to\iota} h) \in \mathcal{H}$ implies $\neg(gw \doteq^{\iota\to\iota} hw) \in \mathcal{H}$ for some new $w \in \Sigma_\iota$ by $\nabla_\mathfrak{f}$. Now, by $\nabla_\exists$ and $\nabla_\vee$, there must be a $p \in \Sigma_{\iota\to o}$ such that $p(gw), \neg p(hw) \in \mathcal{H}$. This contradicts the fact that the only non-atomic formula in Φ (hence in $\mathcal{H}$) is $\neg qg$.

These examples directly lead to the acceptability conditions stated in the next section.

2.2 Acceptable Abstract Consistency Classes

As illustrated by the Examples 2.1.1 and 2.1.2 we need some extra abstract consistency properties to ensure the existence of saturated extensions. These extra properties, which we call *acceptability conditions*, will rule out examples with no saturated extensions as the ones above.

Definition 2.2.1 (Acceptability Conditions) Let Γ_Σ be an abstract consistency class in $\mathfrak{Acc}_*$. We define the following properties:

∇_m If $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$ are atomic and $\mathbf{A}, \neg\mathbf{B} \in \Phi$, then $\Phi * \neg(\mathbf{A} \doteq^o \mathbf{B}) \in \Gamma_\Sigma$.

∇^h_{dec} If $\neg(h\overline{\mathbf{A}^n} \doteq^\beta h\overline{\mathbf{B}^n}) \in \Phi$ for some types α_i where $\beta \in \{o, \iota\}$ and $h_{\overline{\alpha^n}\to\beta} \in \Sigma$, then there is an i with $1 \leq i \leq n$ such that $\Phi * \neg(\mathbf{A}^i \doteq^{\alpha^i} \mathbf{B}^i) \in \Gamma_\Sigma$.

$\nabla_{\xi dec}$ If $h \in \Sigma$ and $\neg(h\overline{\mathbf{A}^n} \doteq^{\beta\to\gamma} h\overline{\mathbf{B}^n}) \in \Phi$ where $h_{\overline{\alpha^n}\to\beta\to\gamma} \in \Sigma$, then there is an i with $1 \leq i \leq n$ such that $\Phi * \neg(\mathbf{A}^i \doteq^{\alpha^i} \mathbf{B}^i) \in \Gamma_\Sigma$.

∇_{fdec} If $\neg(\lambda\overline{X^m}.h\overline{\mathbf{A}^n} \doteq^{\overline{\gamma^m}\to\beta} \lambda\overline{X^m}.h\overline{\mathbf{B}^n}) \in \Phi$ and $h_{\overline{\alpha^i}\to\beta} \in \Sigma \cup \{X^i \mid 0 \leq i \leq m\}$, then there is an i with $1 \leq i \leq n$ such that $\Phi * \neg(\lambda\overline{X^m}.\mathbf{A}^i \doteq^{\overline{\gamma^m}\to\alpha^i} \lambda\overline{X^m}.\mathbf{B}^i) \in \Gamma_\Sigma$.

∇_k If $\neg(\lambda U.\mathbf{A} \doteq^{\alpha\to\beta} \lambda U.\mathbf{B}) \in \Phi$ where $U_\alpha \notin free(\mathbf{A}) \cup free(\mathbf{B})$, then $\Phi * \neg(\mathbf{A} \doteq^\beta \mathbf{B}) \in \Gamma_\Sigma$.

We say Γ_Σ *mates* if it satisfies ∇_m and it *decomposes* if it satisfies ∇^h_{dec} for every parameter $h \in \Sigma$. We say Γ_Σ *logically decomposes* if it satisfies ∇^h_{dec} for every $h \in \Sigma$, including when h is $\neg$, $\vee$, or Π^α for some α.

Finally, Γ_Σ *functionally decomposes* if it satisfies ∇^h_{dec} for every parameter $h \in \Sigma$, as well as properties ∇_{fdec} and ∇_k. It *ξ-decomposes* if it satisfies ∇^h_{dec} for every parameter $h \in \Sigma$ and additionally $\nabla_{\xi dec}$.

Depending on the model class $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$, we will have to assume some combinations of these conditions instead of the saturation condition used in chapter 1.

For there to exist saturated extensions, we require the Γ_Σ in $\mathfrak{Acc}_*$ to satisfy extra conditions. We will summarize these different cases by defining when Γ_Σ is an *acceptable* abstract consistency class in $\mathfrak{Acc}_*$.

Definition 2.2.2 (Acceptable Classes) We define when an abstract consistency class Γ_Σ is *acceptable* in $\mathfrak{Acc}_*$ as follows.

Every Γ_Σ in $\mathfrak{Acc}_{\beta\eta}$ [$\mathfrak{Acc}_\beta$] is *acceptable.*

A Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{b}}$ [$\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$] is *acceptable* if Γ_Σ mates and functionally decomposes.

A Γ_Σ in $\mathfrak{Acc}_{\beta\xi}$ is *acceptable* if Γ_Σ mates, logically decomposes and ξ-decomposes.

A Γ_Σ in $\mathfrak{Acc}_{\beta\xi\mathfrak{b}}$ is *acceptable* if Γ_Σ mates and ξ-decomposes.

A Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{f}}$ is *acceptable* if Γ_Σ mates and logically decomposes.

A Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$ is *acceptable* if Γ_Σ mates and decomposes.

Remark 2.2.3 Throughout the remainder of the paper, we assume Σ is infinite for each type. This implies that every finite set of Σ-sentences is sufficiently Σ-pure.

Definition 2.2.4 (Saturated Extension)
Let $\Gamma_\Sigma \in \mathfrak{Acc}_*$. $\Gamma'_\Sigma \in \mathfrak{Acc}_*$ is a *saturated extension* of Γ_Σ in $\mathfrak{Acc}_*$ if Γ'_Σ is saturated and $\Gamma_\Sigma \subseteq \Gamma'_\Sigma$.

Definition 2.2.5 (Sufficiently Pure Classes)
An abstract consistency class Γ_Σ is *sufficiently Σ-pure* if every $\Phi \in \Gamma_\Sigma$ is sufficiently Σ-pure.

A main result of this paper will be presented as Theorem 2.5.3 in Section 2.5. It is called the **saturated extension theorem** and states:

> *For every model class $\mathfrak{M}_*$, there is a saturated abstract consistency class in $\mathfrak{Acc}_*$ that is an extension of all acceptable sufficiently Σ-pure Γ_Σ in $\mathfrak{Acc}_*$.*

This is by no means a trivial result. In fact, it turns out that the saturated extension theorem implies cut elimination. We will show this relationship to cut elimination as an intermezzo in the next section before we proceed with our formal development.
[2] EdNote(2)

2.3 Saturation and Cut Elimination for Higher Order Sequent Calculi

[3] In this section we discuss the relationship between saturation and cut elimination. For illustration purposes we will investigate the problem with respect to different sequent calculi corresponding to the landscape of model classes introduced in chapter 1. The particular sequent calculi we investigate are $\mathcal{G}_\beta$, $\mathcal{G}_{\beta\eta}$, $\mathcal{G}_{\beta\xi}$, $\mathcal{G}_{\beta\mathfrak{f}}$, $\mathcal{G}_{\beta\mathfrak{b}}$, $\mathcal{G}_{\beta\eta\mathfrak{b}}$, $\mathcal{G}_{\beta\xi\mathfrak{b}}$ and $\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$. They correspond to each notion of abstract consistency $\mathfrak{Acc}_\beta$, $\mathfrak{Acc}_{\beta\eta}$, $\mathfrak{Acc}_{\beta\xi}$, $\mathfrak{Acc}_{\beta\mathfrak{f}}$, $\mathfrak{Acc}_{\beta\mathfrak{b}}$, $\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$, $\mathfrak{Acc}_{\beta\xi\mathfrak{b}}$ and $\mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$. EdNote(3)

[2] EdNote: Chris: Here I commented the Compactification section. Do we need this again in this paper? Why was that here — as a copy identical to the JSL version

[3] EdNote: Chris: Where will we show completeness of the sequent calculi?

$$\frac{\Gamma, (\mathbf{A} \vee \mathbf{B}), \mathbf{A} \to \Delta \quad \Gamma, (\mathbf{A} \vee \mathbf{B}), \mathbf{B} \to \Delta}{\Gamma, (\mathbf{A} \vee \mathbf{B}) \to \Delta} \vee_L \qquad \frac{\Gamma \to (\mathbf{A} \vee \mathbf{B}), \mathbf{A}, \mathbf{B}, \Delta}{\Gamma \to (\mathbf{A} \vee \mathbf{B}), \Delta} \vee_R$$

$$\frac{\Gamma, \neg\mathbf{A} \to \mathbf{A}, \Delta}{\Gamma, \neg\mathbf{A} \to \Delta} \neg_L \qquad \frac{\Gamma, \mathbf{A} \to \neg\mathbf{A}, \Delta}{\Gamma \to \neg\mathbf{A}, \Delta} \neg_R$$

$$\frac{\Gamma, \Pi^\alpha\mathbf{A}, \mathbf{AC} \to \Delta \quad \mathbf{C} \in \mathit{cwff}_\alpha(\Sigma)}{\Gamma, \Pi^\alpha\mathbf{A} \to \Delta} \Pi_L^{\mathbf{C}} \qquad \frac{\Gamma \to \Pi^\alpha\mathbf{A}, \mathbf{A}c, \Delta \quad c_\alpha \in \Sigma \; \mathit{new}}{\Gamma \to \Pi^\alpha\mathbf{A}, \Delta} \Pi_R^c$$

$$\frac{\Gamma, \mathbf{A}, \mathbf{A}{\downarrow_\beta} \to \Delta}{\Gamma, \mathbf{A} \to \Delta} \beta_L \qquad \frac{\Gamma \to \mathbf{A}{\downarrow_\beta}, \mathbf{A}, \Delta}{\Gamma \to \mathbf{A}, \Delta} \beta_R \qquad \frac{}{\Gamma, \mathbf{A} \to \mathbf{A}, \Delta} \mathit{init}$$

Figure 2.1: Higher Order Sequent Calculus $\mathcal{G}_\beta$

Definition 2.3.1 (Sequent) A sequent is of the form $\Gamma \to \Delta$ where Γ and Δ are multisets of sentences from $\mathit{cwff}_o(\Sigma)$.

BegNP(4)

It is convenient to have a notation for a set of sentences Φ obtained from a sequent $\Gamma \to \Delta$.

Definition 2.3.2 (Φ-Sequent)
We let $\Phi_{\Gamma\to\Delta} := \{\mathbf{A}|\mathbf{A} \in \Gamma\} \cup \{\neg\mathbf{B}|\mathbf{B} \in \Delta\}$ and call $\Gamma \to \Delta$ a *Φ-sequent*, iff $\Phi_{\Gamma\to\Delta} \subseteq \Phi$.

Definition 2.3.3 (Valid Sequents) A sequent $\Gamma \to \Delta$ is called *valid for a model* $\mathcal{M}^* \in \mathfrak{M}_*$ where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, if $\mathcal{M}^* \models \mathbf{C}$ for all $\mathbf{C} \in \Gamma$, implies that there is some $\mathbf{D} \in \Delta$ such that $\mathcal{M}^* \models \mathbf{D}$. A sequent $\Gamma \to \Delta$ is called *valid* in $\mathfrak{M}_*$ for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ if it is valid for all models $\mathcal{M}^* \in \mathfrak{M}_*$.

The following statement is a trivial consequence of the above definitions.

Lemma 2.3.4 (Valid Sequents) *A sequent $\Gamma \to \Delta$ is valid for a model $\mathcal{M}^* \in \mathfrak{M}_*$ where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, if $\mathcal{M}^* \models \neg\mathbf{A}$ for some $\mathbf{A} \in \Phi_{\Gamma\to\Delta}$.*

We now define the eight different higher order sequent calculi.

Definition 2.3.5 (Sequent Calculi $\mathcal{G}_*$)
The inference rules defining sequent calculus $\mathcal{G}_\beta$ are shown in Figure 2.1. These rules are included in each $\mathcal{G}_*$. The rules corresponding to forms of extensionality shown in Figure 2.2 will be used to define the different $\mathcal{G}_*$. For completeness, four of the calculi will require various forms and combinations of decomposition rules. These decomposition rules are shown in Figure 2.3.

$\mathcal{G}_{\beta\eta}$: The sequent calculus $\mathcal{G}_{\beta\eta}$ is obtained by adding the rules $\beta\eta_L$ and $\beta\eta_R$ to $\mathcal{G}_\beta$.

$\mathcal{G}_{\beta\mathfrak{b}}$,$\mathcal{G}_{\beta\eta\mathfrak{b}}$: The sequent calculus $\mathcal{G}_{\beta\mathfrak{b}}$ [$\mathcal{G}_{\beta\eta\mathfrak{b}}$] is obtained by adding the rules $\mathit{Init}^{\dot{=}}$, $\mathfrak{b}$, dec^h (for each parameter $h \in \Sigma$), fdec and k to $\mathcal{G}_\beta$ [$\mathcal{G}_{\beta\eta}$].

[4] New Part: reorganized the definition

$\mathcal{G}_{\beta\xi}$: The sequent calculus $\mathcal{G}_{\beta\xi}$ is obtained by adding the rules $Init^{\doteq}$, ξ, ξdec and dec^h (for each $h \in \Sigma$ including when h is $\neg$, $\vee$ or Π^α for some α) to $\mathcal{G}_\beta$.

$\mathcal{G}_{\beta\xi\mathfrak{b}}$: The sequent calculus $\mathcal{G}_{\beta\xi\mathfrak{b}}$ is obtained by adding the rules $Init^{\doteq}$, ξ, b, ξdec and dec^h (for each parameter $h \in \Sigma$) to $\mathcal{G}_\beta$.

$\mathcal{G}_{\beta\mathfrak{f}}$: The sequent calculus $\mathcal{G}_{\beta\mathfrak{f}}$ is obtained by adding the rules $Init^{\doteq}$, f and dec^h (for each $h \in \Sigma$ including when h is $\neg$, $\vee$ or Π^α for some α) to $\mathcal{G}_\beta$.

$\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$: The sequent calculus $\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$ is obtained by adding the rules $Init^{\doteq}$, f, b and dec^h (for each parameter $h \in \Sigma$) to $\mathcal{G}_\beta$.

[5] EdNote(5)

$$\frac{\Gamma, \mathbf{A}, \mathbf{A}{\downarrow_{\beta\eta}} \to \Delta}{\Gamma, \mathbf{A} \to \Delta}\,\beta\eta_L \qquad \frac{\Gamma \to \mathbf{A}{\downarrow_{\beta\eta}}, \mathbf{A}, \Delta}{\Gamma \to \mathbf{A}, \Delta}\,\beta\eta_R$$

$$\frac{\Gamma, \mathbf{A} \to (\mathbf{A} \doteq^o \mathbf{B}), \mathbf{B}, \Delta}{\Gamma, \mathbf{A} \to \mathbf{B}, \Delta}\,Init^{\doteq}$$

$$\frac{\Gamma \to (\forall X_\alpha. \mathbf{M} \doteq^\beta \mathbf{N}), \Delta'}{\Gamma \to (\lambda X_\alpha. \mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha. \mathbf{N}), \Delta}\,\xi \qquad \frac{\Gamma \to (\forall X_\alpha. \mathbf{A}X \doteq^\beta \mathbf{B}X), \Delta'}{\Gamma \to (\mathbf{A} \doteq^{\alpha\to\beta} \mathbf{B}), \Delta}\,f$$

$$\frac{\Gamma, \mathbf{A} \to \mathbf{B}, \Delta' \quad \Gamma, \mathbf{B} \to \mathbf{A}, \Delta'}{\Gamma \to (\mathbf{A} \doteq^o \mathbf{B}), \Delta}\,b$$

$\Delta' \equiv (\mathbf{C} \doteq \mathbf{D}), \Delta$ where $(\mathbf{C} \doteq \mathbf{D})$ is the principal formula of the rule

Figure 2.2: Extensionality Rules for Higher Order Sequent Calculi

We will now discuss the soundness of the calculi $\mathcal{G}_*$, i.e., for $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ we will show that sequent calculus $\mathcal{G}_*$ is sound for model class $\mathfrak{M}_*$.

Theorem 2.3.6 (Soundness)
$\mathcal{G}_*$ is sound for $\mathfrak{M}_*$ where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. That is, if $\Vdash_{\mathcal{G}_*} \Gamma \to \Delta$ is derivable, then the sequent $\Gamma \to \Delta$ is valid in $\mathfrak{M}_*$.

Proof: This can be shown by induction on the derivation of $\Vdash_{\mathcal{G}_*} \Gamma \to \Delta$. We distinguish based on the last rule of the derivation. The only base case is *init*. Since $\mathbf{A}, \neg\mathbf{A} \in \Phi_{\Gamma, \mathbf{A} \to \mathbf{A}, \Delta}$ and for each model $\mathcal{M}^* \in \mathfrak{M}_*$ we have $\mathcal{M}^* \models \neg\mathbf{A}$ or $\mathcal{M}^* \models \neg\neg\mathbf{A}$, we trivially get the assertion with Lemma 2.3.4.

[5] EdNote: Chris: In this sequent calculus we oprate with $\mathbf{A}{\downarrow_{\beta\eta}}$ while in the ND calculus in the other paper we operate with $\equiv_{\beta\eta}$. Should we also use the latter here? Just to harmonize things?

$$\frac{\Gamma \to (\mathbf{A}^1 \doteq^{\alpha_1} \mathbf{B}^1), \Delta' \quad \cdots \quad \Gamma \to (\mathbf{A}^n \doteq^{\alpha_n} \mathbf{B}^n), \Delta' \quad n \geq 0, \beta \in \{o, \iota\}, h_{\overline{\alpha^n} \to \beta} \in \Sigma}{\Gamma \to (h\overline{\mathbf{A}^n} \doteq^{\beta} h\overline{\mathbf{B}^n}), \Delta} \, dec^h$$

$$\frac{\Gamma \to (\mathbf{A}^1 \doteq^{\alpha_1} \mathbf{B}^1), \Delta' \quad \cdots \quad \Gamma \to (\mathbf{A}^n \doteq^{\alpha_n} \mathbf{B}^n), \Delta' \quad n \geq 0, h_{\overline{\alpha^n} \to \beta \to \gamma} \in \Sigma}{\Gamma \to (h\overline{\mathbf{A}^n} \doteq^{\beta \to \gamma} h\overline{\mathbf{B}^n}), \Delta} \, \xi dec$$

$$\frac{\Gamma \to (\lambda \overline{X^m}.\mathbf{A}^1 \doteq^{\overline{\gamma^m} \to \alpha_1} \lambda \overline{X^m}.\mathbf{B}^1), \Delta' \quad \cdots \quad \Gamma \to (\lambda \overline{X^m}.\mathbf{A}^n \doteq^{\overline{\gamma^m} \to \alpha_n} \lambda \overline{x^m}.\mathbf{B}^n), \Delta' \quad **}{\Gamma \to (\lambda \overline{X^m}.h\overline{\mathbf{A}^n} \doteq^{\overline{\gamma^m} \to \beta} \lambda \overline{X^m}.h\overline{\mathbf{B}^n}), \Delta} \, fdec$$

$$\frac{\Gamma \to \mathbf{A} \doteq^{\beta} \mathbf{B}, \Delta' \quad U_\alpha \notin free(\mathbf{A}) \cup free(\mathbf{B})}{\Gamma \to (\lambda U.\mathbf{A} \doteq^{\alpha \to \beta} \lambda U.\mathbf{B}), \Delta} \, k$$

$** \equiv m, n \geq 0, h_{\overline{\alpha^n} \to \beta} \in \Sigma \cup \overline{X^m}$
$\Delta' \equiv (\mathbf{C} \doteq \mathbf{D}), \Delta$ where $(\mathbf{C} \doteq \mathbf{D})$ is the principal formula of the rule

Figure 2.3: Decomposition Rules for Higher Order Sequent Calculi

$\vee_L$: By induction $\Gamma, (\mathbf{A} \vee \mathbf{B}), \mathbf{A} \to \Delta$ and $\Gamma, (\mathbf{A} \vee \mathbf{B}), \mathbf{B} \to \Delta$ are valid sequents for $\mathcal{M}^* \in \mathfrak{M}_*$. Hence $\mathcal{M}^* \models \mathbf{C}$ for all $\mathbf{C} \in \Gamma, (\mathbf{A} \vee \mathbf{B}), \mathbf{A}$ implies $\mathcal{M}^* \models \mathbf{D}$ for some $\mathbf{D} \in \Delta$, and $\mathcal{M}^* \models \mathbf{C}'$ for all $\mathbf{C}' \in \Gamma, (\mathbf{A} \vee \mathbf{B}), \mathbf{B}$ implies $\mathcal{M}^* \models \mathbf{D}'$ for some $\mathbf{D}' \in \Delta$. Now let $\mathcal{M}^* \in \mathfrak{M}_*$ such that $\mathcal{M}^* \models \mathbf{C}''$ for all $\mathbf{C}'' \in \Gamma, (\mathbf{A} \vee \mathbf{B})$. We get $\mathcal{M}^* \models \mathbf{A}$ or $\mathcal{M}^* \models \mathbf{B}$, and in either case $\mathcal{M}^* \models \mathbf{D}''$ for some $\mathbf{D}'' \in \Delta$ by induction hypothesis.

$\vee_R$: We easily get the assertion by induction if we show that $\mathcal{M}^* \models \mathbf{A}$ or $\mathcal{M}^* \models \mathbf{B}$ whenever $\mathcal{M}^* \models \mathbf{A} \vee \mathbf{B}$. This follows trivially.

$\neg_L, \neg_R$: These cases follow by a simple case distinction employing the fact that $\mathcal{M}^* \models \mathbf{A}$ or $\mathcal{M}^* \models \neg\mathbf{A}$ for each $\mathcal{M}^* \in \mathfrak{M}_*$.

$\Pi_L^{\mathbf{C}}$: We easily get the assertion by induction if we show that $\mathcal{M}^* \models \Pi^\alpha \mathbf{A}$ implies $\mathcal{M}^* \models \mathbf{A}\mathbf{C}$. This follows trivially.

Π_R^c: We easily get the assertion by induction if we show that $\mathcal{M}^* \models \Pi\mathbf{A}$ whenever $\mathcal{M}^* \models \mathbf{A}c$ for any parameter c that does not occur in $\Gamma, \Pi^\alpha \mathbf{A}$ or Δ. Let $\mathcal{M}^* \models \mathbf{A}c$ for any such parameter c and assume $\mathcal{M}^* \not\models \Pi\mathbf{A}$. Then there must exists some $\mathsf{a} \in \mathcal{D}_\alpha$ such that $v(\mathcal{E}(\mathbf{A})@\mathsf{a}) \equiv \mathtt{F}$, i.e., for any assignment φ and fresh variable X_α not occuring on $\mathbf{A}$ we have $\mathcal{M}^* \not\models_{\varphi, [\mathsf{a}/X]} (\mathbf{A}X)$ and hence $\mathcal{M}^* \models_{\varphi, [\mathsf{a}/X]} \neg(\mathbf{A}X)$. By Lemma 2.6.2 we know that there exists a model $M^{*'} \equiv (\mathcal{D}, @, \mathcal{E}', v)$ such that $\mathcal{E}'(c) \equiv \mathsf{a}$ and $\mathcal{M}^{*'} \models_\varphi [c/X]\neg(\mathbf{A}X)$. Hence, $\mathcal{M}^{*'} \not\models_\varphi [c/X](\mathbf{A}X)$. Since X is fresh we thus have $\mathcal{M}^{*'} \not\models \mathbf{A}c$. We know get a contradiction with Lemma 2.6.2 and $\mathcal{M}^* \models \mathbf{A}c$.

β_L, β_R: These cases follow trivially since Σ-evaluations and thus Σ-models respect β-reduction by Definition 1.3.17(4).

We now check soundness of the rules $Init^{\doteq}, \xi dec, dec^h, fdec$, and k in Figure 2.2 and Figure 2.3. Like the rules above they are sound for all model classes.

$Init^{\doteq}$**:** We easily get the assertion by induction if we show that $\mathcal{M}^* \models \mathbf{B}$ whenever $\mathcal{M}^* \models \mathbf{A}$ and $\mathcal{M}^* \models \mathbf{A} \doteq^o \mathbf{B}$. Let assignment φ be arbitrary. We get from the latter that for all $p \in \mathcal{D}_{o\to o}$ holds $\upsilon(p@\mathcal{E}_\varphi(\mathbf{A})) \equiv \mathtt{F}$ or $\upsilon(p@\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{T}$. In particular for Leibniz property $p := \mathcal{E}_\varphi(\lambda X.X)$ we get $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) \equiv \mathtt{F}$ or $\upsilon(\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{T}$. Since we have $\upsilon(\mathcal{E}_\varphi(\mathbf{A})) \equiv \mathtt{T}$ by $\mathcal{M}^* \models \mathbf{A}$ we know that $\upsilon(\mathcal{E}_\varphi(\mathbf{B})) \equiv \mathtt{T}$.

k**:** We easily get the assertion by induction if we show that that $\mathcal{M}^* \models \lambda U_\alpha.\mathbf{A} \doteq^{\alpha\to\beta} \lambda U_\alpha.\mathbf{B}$ for $U \notin free(\mathbf{A}) \cup free(\mathbf{B})$ whenever $\mathcal{M}^* \models \mathbf{A} \doteq^\beta \mathbf{B}$, which entails the assertion. Let assignment φ be arbitrary. Choosing Leibniz property $p := \mathcal{E}_\varphi(\lambda X_{\alpha\to\beta}.(\lambda U.\mathbf{A}) \doteq (\lambda U.X))$ and employing the fact that $U \notin free(\mathbf{A}) \cup free(\mathbf{B})$ we can analogously to above show that $\upsilon(\mathcal{E}_\varphi((\lambda U.\mathbf{A}) \doteq (\lambda U.\mathbf{A}))) \equiv \mathtt{F}$ or $\upsilon(\mathcal{E}_\varphi((\lambda U.\mathbf{A}) \doteq (\lambda U.\mathbf{B}))) \equiv \mathtt{T}$. Since Leibniz equality is reflexive by Lemma 1.4.2(1) we rule out the former option and get $\upsilon(\mathcal{E}_\varphi(\lambda U.\mathbf{A}) \doteq (\lambda U.\mathbf{B})) \equiv \mathtt{T}$. Hence $\mathcal{M}^* \models (\lambda U.\mathbf{A}) \doteq (\lambda U.\mathbf{B})$.

dec^h**:** We easily get the assertion by induction if we show that $\mathcal{M}^* \models h\overline{\mathbf{A}} \doteq^\alpha h\overline{\mathbf{B}}$ whenever $\mathcal{M}^* \models \mathbf{A}_i \doteq^{\alpha_i} \mathbf{B}_i$ for $i \equiv 1,\ldots,n$. Let assignment φ be arbitrary. Choosing Leibniz property $p_1 := \mathcal{E}_\varphi(\lambda X_{\alpha_i}.h\mathbf{A}_1 \doteq hX)$ we can analogously to above derive from $\mathcal{M}^* \models \mathbf{A}_1 \doteq^{\alpha_i} \mathbf{B}_1$ that $\mathcal{M}^* \models h\mathbf{A}_1 \doteq h\mathbf{B}_1$ holds. Next we subsequently apply the Leibniz equations $\mathbf{A}_i \doteq^{\alpha_i} \mathbf{B}_i$ ($i \equiv 2,\ldots,n$) for $n-1$ times. We illustrate the iterative process only for the first step, since the remaining steps are analogous. From $\mathcal{M}^* \models h\mathbf{A}_1 \doteq h\mathbf{B}_1$ we get with Leibniz property $p_{2'} := \mathcal{E}_\varphi(\lambda X_{\alpha_2\to\ldots\to\alpha_n\to o}.(h\mathbf{A}_1)\mathbf{A}_2 \doteq X\mathbf{A}_2)$ that $\mathcal{M}^* \models (h\mathbf{A}_1)\mathbf{A}_2 \doteq (h\mathbf{B}_1)\mathbf{A}_2$. From $\mathcal{M}^* \models \mathbf{A}_2 \doteq^{\alpha_i} \mathbf{B}_2$ we similarly get with Leibniz property $p_{2''} := \mathcal{E}_\varphi(\lambda X._{\alpha_i} h(\mathbf{B}_1)\mathbf{A}_2 \doteq (h\mathbf{B}_1)X)$ that $\mathcal{M}^* \models (h\mathbf{B}_1)\mathbf{A}_2 \doteq (h\mathbf{B}_1)\mathbf{B}_2$. Lemma 1.4.2[6] By transitivity of Leibniz equality, which EdNote(6) is consequence of Lemma 1.4.2[7], we get $\mathcal{M}^* \models (h\mathbf{A}_1)\mathbf{A}_2 \doteq (h\mathbf{B}_1)\mathbf{B}_2$. EdNote(7)

ξdec**:** The proof is analogous to dec^h. [8] EdNote(8)

$fdec$**:** The proof is analogous to dec^h with $p_1 := \mathcal{E}_\varphi(\lambda Z.(\lambda\overline{X}.h\mathbf{A}_1) \doteq (\lambda\overline{X}.h(Z\overline{X})))$, $p_{2'} := \lambda Z.(\lambda\overline{X}.(h\mathbf{A}_1)\mathbf{A}_2) \doteq (\lambda\overline{X}.(Z\overline{X})\mathbf{A}_2)$ and $p_{2''} := \lambda Z.(\lambda\overline{X}.(h\mathbf{B}_1)\mathbf{A}_2) \doteq (\lambda\overline{X}.(h\mathbf{B}_1)(Z\overline{X}))$ as instances of the first few Leibniz properties to be employed.

We now check soundness of the remaining rules in Figure 2.2 and Figure 2.3, which are $\beta\eta_L$, $\beta\eta_R$, ξ, f and b, with respect to their model classes.

$\beta\eta_L$, $\beta\eta_R$**:** The rules $\beta\eta_L$ and $\beta\eta_R$ only apply to model classes where property η holds. By Definition of property η (see 1.3.46, 1.3.41 and 1.3.19) we know that all Σ-models $\mathcal{M}^{*\eta} \in \mathfrak{M}_{*\eta}$ for $*\eta \in \{\beta\eta, \beta\mathfrak{f}, \beta\eta\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ respect η-reduction.

ξ**:** Rule ξ only applies to model classes where property ξ holds. We easily get the assertion by induction if we show that for any $\mathcal{M}^{*\xi} \in \mathfrak{M}_{*\xi}$ where $*\xi \in \{\beta\xi, \beta\mathfrak{f}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ $\mathcal{M}^{*\xi} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon) \models \lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha.\mathbf{N}$ whenever $\mathcal{M}^{*\xi} \models \forall X_\alpha.\mathbf{M} \doteq^\beta \mathbf{N}$. From the latter we know for all $a \in \mathcal{D}_\alpha$ that for any assignment φ and any $\mathsf{a} \in \mathcal{D}_\alpha$ we have $\mathcal{M}^{*\xi} \models_{\varphi,[a/X]} \mathbf{M} \doteq^\beta \mathbf{N}$. Then by Lemma 1.4.2(2) $\mathcal{E}_{\varphi,[a/X]}\mathbf{M} \equiv \mathcal{E}_{\varphi,[a/X]}\mathbf{N}$. Employing the definition of ξ (see 1.3.46, 1.3.41 and 1.3.19) we conclude $\mathcal{E}_\varphi(\lambda X_\alpha.\mathbf{M}) \equiv \mathcal{E}_\varphi(\lambda X_\alpha.\mathbf{N})$. Then $\mathcal{M}^{*\xi} \models \lambda X_\alpha.\mathbf{M} \doteq^{\alpha\to\beta} \lambda X_\alpha.\mathbf{N}$ follows by Lemma 1.4.2(1).

[6] EdNote: Chris: or do we have we have a better reference?
[7] EdNote: Chris: or do we have we have a better reference?
[8] EdNote: Chad: added this to the list. still need to check.

$\mathfrak{f}$: Rule ξ only applies to model classes where property $\mathfrak{f}$ holds. We easily get the assertion by induction if we show that for any $\mathcal{M}^{*\mathfrak{f}} \in \mathfrak{M}_{*\mathfrak{f}}$ where $* \in \{\beta\mathfrak{f}, \beta\mathfrak{f}\mathfrak{b}\}$ $\mathcal{M}^{*\mathfrak{f}} \models \mathbf{A} \doteq^{\alpha\to\beta} \mathbf{B}$ whenever $\mathcal{M}^{*\xi} \models \forall X_\alpha.\mathbf{A}X \doteq^{\beta} \mathbf{B}X$, which entails the assertion. The argumentation is similar to above using the definition of $\mathfrak{f}$ (see 1.3.46, 1.3.41 and 1.3.19).

$\mathfrak{b}$: We easily get the assertion by induction if we show that for any $\mathcal{M}^* \in \mathfrak{M}_*$ where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$ that $\mathcal{M}^* \models \mathbf{A} \doteq^o \mathbf{B}$ whenever $\mathcal{M}^* \models \mathbf{A}$ implies that $\mathcal{M}^* \models \mathbf{B}$ and vice versa. From the latter we can conclude that $\mathcal{M}^* \models \mathbf{A} \Leftrightarrow \mathbf{B}$. Since property $\mathfrak{b}$ holds we also know by Lemma 1.4.3(4) that $\mathcal{M}^* \models (\mathbf{A} \Leftrightarrow \mathbf{B}) \Rightarrow (\mathbf{A} \doteq \mathbf{B})$. As a simple consequence we get $\mathcal{M}^* \models (\mathbf{A} \doteq \mathbf{B})$.

□

Note that in the weaker extensional systems $\mathcal{G}_{\beta\mathfrak{b}}$, $\mathcal{G}_{\beta\eta\mathfrak{b}}$, $\mathcal{G}_{\beta\xi}$, $\mathcal{G}_{\beta\xi\mathfrak{b}}$ and $\mathcal{G}_{\beta\mathfrak{f}}$, we needed quite elaborate decomposition rules. The need for these decomposition rules is illustrated by the following examples.

Example 2.3.7 In $\mathcal{G}_{\beta\mathfrak{b}}$, we should be able to derive the sequent $a_o, b_o \to (\lambda U_\alpha.a \doteq^o \lambda U_\alpha.b)$ since we can easily prove $(a \doteq^o b) \to (\lambda U_\alpha.a \doteq^o \lambda U_\alpha.b)$ in $\mathcal{G}_\beta$. This indicates the need for rule k. Similarly, we should be able to derive the sequents $a_o, b_o \to ((\lambda U_\alpha.\neg a) \doteq^o (\lambda U_\alpha.\neg b))$ and $a_o, b_o \to ((\lambda X_{o\to o}.(Xa)) \doteq^o (\lambda X_{o\to o}.(Xb)))$, indicating the need for $fdec$. In any of these examples, we can derive the sequent in $\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$ without $fdec$ and k by making use of f.

Example 2.3.8 In $\mathcal{G}_{\beta\mathfrak{f}}$, we should be able to derive the sequent $\forall X_\alpha.((f_{\alpha\to\beta}X) \doteq^\beta (g_{\alpha\to\beta}X)) \to (\neg(p_{(\alpha\to\beta)\to o}f)) \doteq^o (\neg(p_{(\alpha\to\beta)\to o}g))$ This indicates the need for dec^h where h is a logical constant. Note that in $\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$ we could use rule b to eliminate the need for logical decomposition.

Remark 2.3.9 It is easy to verify by induction on derivations that each sequent calculus $\mathcal{G}_*$ satisfies the usual weakening and contraction properties.

We now introduce abstract consistency classes for each sequent calculus $\mathcal{G}_*$.

Definition 2.3.10 (Abstract Consistency Class for $\mathcal{G}_*$)
For each sequent calculus $\mathcal{G}_*$ where $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$, let $\Gamma_\Sigma^{\mathcal{G}_*}$ be the class of all sufficiently Σ-pure $\Phi \subseteq cwff_o(\Sigma)$ such that no Φ-sequent is $\mathcal{G}_*$-derivable.[9] EdNote(9)

We must verify $\Gamma_\Sigma^{\mathcal{G}_*}$ is an abstract consistency class,[10] which is going to be easy, since our sequent calculi were specifically designed for this. EndNP(4) EdNote(10)

Lemma 2.3.11 *$\Gamma_\Sigma^{\mathcal{G}_*}$ is in $\mathfrak{Acc}_*$.*

Proof: First, *Init* assures ∇_c. The verification of the other conditions follow the same basic outline. Suppose $(\mathbf{A} \lor \mathbf{B}) \in \Phi$, $\Phi \cup \{\mathbf{A}\} \notin \Gamma_\Sigma^{\mathcal{G}_*}$ and $\Phi \cup \{\mathbf{B}\} \notin \Gamma_\Sigma^{\mathcal{G}_*}$. By definition of $\Gamma_\Sigma^{\mathcal{G}_*}$, there must be $\mathcal{G}_*$-derivations of $\mathbf{A}, \Gamma_1 \to \Delta_1$ and $\mathbf{B}, \Gamma_2 \to \Delta_2$ for some Φ-sequents $\Gamma_1 \to \Delta_1$ and $\Gamma_2 \to \Delta_2$. By weakening, there are $\mathcal{G}_*$-derivations of $\mathbf{A}, (\mathbf{A} \lor \mathbf{B}), \Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2$ and $\mathbf{B}, (\mathbf{A} \lor \mathbf{B}), \Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2$. The $\lor_L$ rule allows us to derive the Φ-sequent $(\mathbf{A} \lor \mathbf{B}), \Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2$, showing $\Phi \notin \Gamma_\Sigma^{\mathcal{G}_*}$. Hence $\nabla_\lor$ holds. By similar arguments, $\lor_R$ implies $\nabla_\land$, Π_L implies $\nabla_\forall$, Π_R implies $\nabla_\exists$, $\neg_L$ and $\neg_R$

[9] EdNote: Chad: I'm the one who included "suff pure" in the defn of the acc, but now I'm not sure that's what we really want.
[10] EdNote: they are not compact, because of suff purity

imply $\nabla_\neg$, and β_L and β_R imply ∇_β. For $\mathcal{G}_{\beta\eta}$ and $\mathcal{G}_{\beta\eta\mathfrak{b}}$, the rules $\beta\eta_L$ and $\beta\eta_R$ imply ∇_η. For $\mathcal{G}_{\beta\mathfrak{b}}$, $\mathcal{G}_{\beta\eta\mathfrak{b}}$ and $\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$, the rule b implies $\nabla_\mathfrak{b}$. For $\mathcal{G}_{\beta\xi}$ and $\mathcal{G}_{\beta\xi\mathfrak{b}}$, the rule ξ implies ∇_ξ. For $\mathcal{G}_{\beta\mathfrak{f}}$ and $\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$, the rule f implies $\nabla_\mathfrak{f}$. □

Now we are in the position to illustrate the relationship between Saturation and cut elimination.

Remark 2.3.12 (Cut Elimination and Saturation) Suppose we try to show $\Gamma_\Sigma^{\mathcal{G}_*}$ is saturated. Let $\Phi \subseteq \mathit{cwff}_o(\Sigma)$ and $\mathbf{C} \in \mathit{cwff}_o(\Sigma)$ where $\Phi{*}\mathbf{C} \notin \Gamma_\Sigma^{\mathcal{G}_*}$ and $\Phi{*}\neg\mathbf{C} \notin \Gamma_\Sigma^{\mathcal{G}_*}$. Then (via weakening) for some Φ-sequent $\Gamma \to \Delta$, we have $\mathcal{G}_*$-derivations of $\Gamma \to \mathbf{C}, \Delta$ and $\Gamma, \mathbf{C} \to \Delta$. If cut is admissible, then there is a derivation of $\Gamma \to \Delta$. This shows $\Phi \notin \Gamma_\Sigma^{\mathcal{G}_*}$, so $\Gamma_\Sigma^{\mathcal{G}_*}$ is saturated. Below, we will use the saturation extension theorem to show cut is admissible, so that each $\Gamma_\Sigma^{\mathcal{G}_*}$ is, in fact, saturated.

Conversely, suppose we know $\Gamma_\Sigma^{\mathcal{G}_*}$ is saturated and we have $\mathcal{G}_*$-derivations of $\Gamma \to \mathbf{C}, \Delta$ and $\Gamma, \mathbf{C} \to \Delta$. The derivations witness $\Phi_{\Gamma\to\Delta} * \mathbf{C} \notin \Gamma_\Sigma^{\mathcal{G}_*}$ and $\Phi_{\Gamma\to\Delta} * \neg\mathbf{C} \notin \Gamma_\Sigma^{\mathcal{G}_*}$. By saturation, $\Phi_{\Gamma\to\Delta} \notin \Gamma_\Sigma^{\mathcal{G}_*}$. But (since $\Phi_{\Gamma\to\Delta}$ is finite, hence sufficiently pure) this implies there is a $\mathcal{G}_*$-derivation of $\Gamma \to \Delta$. That is, saturation of $\Gamma_\Sigma^{\mathcal{G}_*}$ implies cut is admissible.

This argument demonstrates that showing saturation directly for a calculus can be as difficult as showing cut elimination. In classical higher order logic, cut elimination has only been shown by semantic methods. In fact, strong proof methods are necessary to prove cut elimination, since cut elimination implies consistency of analysis.[11] EdNote(11)

In fact, we we will show below that the saturation extension theorem implies cut elimination for each $\mathcal{G}_*$. This implies that the proof of the saturated extension theorem must require methods at least as strong as those required for classical higher order cut elimination.

Next we will prove a kind of soundness result for saturatedness. By definition Φ-sequents are not $\mathcal{G}_*$-derivable for $\Phi \in \Gamma_\Sigma^{\mathcal{G}_*}$. This condition also hols for any saturated extension of $\Gamma_\Sigma^{\mathcal{G}_*}$.

Lemma 2.3.13 *Suppose Γ'_Σ is a saturated extension of $\Gamma_\Sigma^{\mathcal{G}_*}$. If $\mathcal{D}$ is a $\mathcal{G}^*$-derivation of $\Gamma \to \Delta$, then $\Phi_{\Gamma\to\Delta} \notin \Gamma'_\Sigma$.*

Proof: We prove this by induction on the derivation $\mathcal{D}$. We distinguish over the last rule applied. The base case consists only of the *Init* and in this case there is an $\mathbf{A} \in \Gamma \cap \Delta$, so $\mathbf{A}, \neg\mathbf{A} \in \Phi_{\Gamma\to\Delta} \notin \Gamma'^*_\Sigma$ by ∇_c.

$\vee_L$: In this case, the active formula is some $\mathbf{A} \vee \mathbf{B} \in \Gamma$. By induction, we know $\Phi_{\Gamma,\mathbf{A}\to\Delta} \equiv \Phi_{\Gamma\to\Delta} * \mathbf{A} \notin \Gamma'_\Sigma$ and $\Phi_{\Gamma,\mathbf{B}\to\Delta} \equiv \Phi_{\Gamma\to\Delta} * \mathbf{B} \notin \Gamma'_\Sigma$. By $\nabla_\vee$, $\Phi_{\Gamma\to\Delta} \notin \Gamma'_\Sigma$.

$\vee_R$: In this case, the active formula is some $\mathbf{A} \vee \mathbf{B} \in \Delta$. By induction, we know $\Phi_{\Gamma\to\mathbf{A},\mathbf{B},\Delta} \equiv \Phi_{\Gamma\to\Delta} \cup \{\neg\mathbf{A}, \neg\mathbf{B}\} \notin \Gamma'_\Sigma$. By $\nabla_\wedge$, $\Phi_{\Gamma\to\Delta} \notin \Gamma'_\Sigma$.

$\neg_L$: The active formula is some $\neg\mathbf{A} \in \Gamma$. By induction, we know $\Phi_{\neg\mathbf{A},\Gamma\to\mathbf{A},\Delta} \equiv \Phi_{\Gamma\to\Delta} * \neg\mathbf{A} \equiv \Phi_{\neg\mathbf{A},\Gamma\to\Delta} \notin \Gamma'_\Sigma$.

$\neg_R$: Analogous to $\vee_R$, this case follows from $\nabla_\neg$.

$\Pi_L^{\mathbf{C}}$: Analogous to $\vee_R$, this case follows from $\nabla_\forall$.

Π_R^c: Analogous to $\vee_R$, this case follows from $\nabla_\exists$, and the fact that Σ is infinite at each type.

β_L, β_R: Analogous to $\vee_R$, these cases follow from ∇_β.

[11] EdNote: reference Takeuti and Andrews

$\beta\eta_L, \beta\eta_R$**:** If this rule occurs, $\mathcal{G}_*$ must be $\mathcal{G}_{\beta\eta}$ or $\mathcal{G}_{\beta\eta\mathfrak{b}}$. In these cases, Γ'_Σ is a $\mathfrak{Acc}_*$ satisfying ∇_η. Now analogous to $\vee_R$, this case follows from ∇_η.

Init$^{\doteq}$**:** By induction, we have $\Phi_{\Gamma\to\Delta} * \neg(\mathbf{A} \doteq^o \mathbf{B}) \notin \Gamma'_\Sigma$ where $\mathbf{A} \in \Gamma$ and $\mathbf{B} \in \Delta$. Assume $\Phi_{\Gamma\to\Delta} \in \Gamma'_\Sigma$. Since Γ'_Σ is saturated, we must have $\Phi_{\Gamma\to\Delta} * (\mathbf{A} \doteq^o \mathbf{B}) \in \Gamma'_\Sigma$. Applying $\nabla_\forall$ with $\lambda X_o.X$ and ∇_β, we have $\Phi_{\Gamma\to\Delta} \cup \{(\mathbf{A} \doteq^o \mathbf{B}), \neg\mathbf{A} \vee \mathbf{B}\} \in \Gamma'_\Sigma$. By $\nabla_\vee$, we must have $\Phi_{\Gamma\to\Delta} \cup \{(\mathbf{A} \doteq^o \mathbf{B}), \neg\mathbf{A} \vee \mathbf{B}, \neg\mathbf{A}\} \in \Gamma'_\Sigma$ or $\Phi_{\Gamma\to\Delta} \cup \{(\mathbf{A} \doteq^o \mathbf{B}), \neg\mathbf{A} \vee \mathbf{B}, \mathbf{B}\} \in \Gamma'_\Sigma$, but this contradicts $\mathbf{A}, \neg\mathbf{B} \in \Phi_{\Gamma\to\Delta}$.[12] EdNote(12)

dec^h**:** By induction, we have $\Phi_{\Gamma\to\Delta} * \neg(\mathbf{A}^i \doteq \mathbf{B}^i) \notin \Gamma'_\Sigma$ for $1 \leq i \leq n$ where $(h\overline{\mathbf{A}^n} \doteq h\overline{\mathbf{B}^n}) \in \Delta$. Assume $\Phi_{\Gamma\to\Delta} \in \Gamma'_\Sigma$. Since Γ'_Σ is saturated, $\Phi_{\Gamma\to\Delta} \cup \{\mathbf{A}^1 \doteq \mathbf{B}^1, \ldots, \mathbf{A}^n \doteq \mathbf{B}^n\} \in \Gamma'_\Sigma$. We can use $\nabla_\forall$ and ∇_β (n times) [13] to conclude $\Phi_{\Gamma\to\Delta} \cup \{\mathbf{A}^1 \doteq \mathbf{B}^1, \ldots, \mathbf{A}^n \doteq \mathbf{B}^n, (h\overline{\mathbf{A}^n} \doteq h\overline{\mathbf{B}^n})\} \in \Gamma'_\Sigma$ EdNote(13) which contradicts $\neg(h\overline{\mathbf{A}^n} \doteq h\overline{\mathbf{B}^n}) \in \Phi_{\Gamma\to\Delta}$.[14] EdNote(14)

fdec: By induction, we have $\Phi_{\Gamma\to\Delta} * \neg(\lambda\overline{X^m}.\mathbf{A}^i \doteq \lambda\overline{X^m}.\mathbf{B}^i) \notin \Gamma'_\Sigma$ for $1 \leq i \leq n$ where $(\lambda\overline{X^m}.h\overline{\mathbf{A}^n}) \doteq (\lambda\overline{X^m}.h\overline{\mathbf{B}^n}) \in \Delta$. Assume $\Phi_{\Gamma\to\Delta} \in \Gamma'_\Sigma$. Since Γ'_Σ is saturated, we must have $\Phi_{\Gamma\to\Delta} \cup \{\lambda\overline{X^m}.\mathbf{A}^1 \doteq \lambda\overline{X^m}.\mathbf{B}^1, \ldots, \lambda\overline{X^m}.\mathbf{A}^n \doteq \lambda\overline{X^m}.\mathbf{B}^n\} \in \Gamma'_\Sigma$. [15] Using $\nabla_\forall$ (n times) with $\lambda G.(\lambda\overline{X^m}.h\overline{\mathbf{A}^n} \doteq$ EdNote(15) $(\lambda\overline{X^m}.h\overline{\mathbf{A}^{j-1}}(G\overline{X^m})\mathbf{B}^{j+1}\ldots\mathbf{B}^n)$ for each $1 \leq j \leq n$, we can find a $\Phi \in \Gamma'_\Sigma$ with $\Phi_{\Gamma\to\Delta} \cup \{\lambda\overline{X^m}.\mathbf{A}^1 \doteq \lambda\overline{X^m}.\mathbf{B}^1, \ldots, \lambda\overline{X^m}.\mathbf{A}^n \doteq \lambda\overline{X^m}.\mathbf{B}^n\} \subseteq \Phi$ and $\lambda\overline{X^m}.h\overline{\mathbf{A}^n} \doteq \lambda\overline{X^m}.h\overline{\mathbf{B}^n} \in \Phi$. This contradicts $\neg(\lambda\overline{X^m}.h\overline{\mathbf{A}^n} \doteq \lambda\overline{X^m}.h\overline{\mathbf{B}^n}) \in \Phi_{\Gamma\to\Delta}$. [16] EdNote(16)

k**:** By induction, we have $\Phi_{\Gamma\to\Delta} * \neg(\mathbf{A} \doteq \mathbf{B}) \notin \Gamma'_\Sigma$ where $\lambda U.\mathbf{A} \doteq \lambda U.\mathbf{B} \in \Delta$ and $U \notin free(\mathbf{A}) \cup free(\mathbf{B})$. Assume $\Phi_{\Gamma\to\Delta} \in \Gamma'_\Sigma$. Then by saturation, we must have $\Phi_{\Gamma\to\Delta} * \mathbf{A} \doteq \mathbf{B} \in \Gamma'_\Sigma$. Applying $\nabla_\forall$ with $\lambda Y.((\lambda U.\mathbf{A}) \doteq (\lambda U.Y))$, we can obtain a $\Phi \in \Gamma'_\Sigma$ with $\Phi_{\Gamma\to\Delta} * \mathbf{A} \doteq \mathbf{B} \subseteq \Phi$ and $\lambda U.\mathbf{A} \doteq \lambda U.\mathbf{B} \in \Phi$ contradicting $\neg(\lambda U.\mathbf{A} \doteq \lambda U.\mathbf{B}) \in \Phi_{\Gamma\to\Delta}$.[17] EdNote(17)

b**:** If this rule occurs, $\mathcal{G}_*$ must be $\mathcal{G}_{\beta\mathfrak{b}}$, $\mathcal{G}_{\beta\eta\mathfrak{b}}$ or $\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$. So, Γ'_Σ is a $\mathfrak{Acc}_*$ satisfies $\nabla_\mathfrak{b}$ which is used to prove this case.

f**:** If this rule occurs, $\mathcal{G}_*$ must be $\mathcal{G}_{\beta\mathfrak{f}}$ or $\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$. So, Γ'_Σ is a $\mathfrak{Acc}_*$ satisfies $\nabla_\mathfrak{f}$ which is used to prove this case. □

Theorem 2.3.14 (Cut Elimination) *Cut is admissible* $\mathcal{G}_*$, *if* $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$.

Proof: In each case, we defined $\Gamma^{\mathcal{G}_*}_\Sigma$ to only contain sufficiently Σ-pure sets. We verify that each $\Gamma^{\mathcal{G}_*}_\Sigma$ is acceptable:

β**,** $\beta\eta$ We always have $\Gamma^{\mathcal{G}_\beta}_\Sigma$ and $\Gamma^{\mathcal{G}_{\beta\eta}}_\Sigma$ are acceptable.

$\beta\mathfrak{b}$**,** $\beta\eta\mathfrak{b}$ For $\mathcal{G}_{\beta\mathfrak{b}}$ and $\mathcal{G}_{\beta\eta\mathfrak{b}}$, the rules *Init*$^{\doteq}$, *dec*^{h} (for parameter h), *fdec* and k imply $\Gamma^{\mathcal{G}_{\beta\mathfrak{b}}}_\Sigma$ and $\Gamma^{\mathcal{G}_{\beta\eta\mathfrak{b}}}_\Sigma$ mate and functionally decompose. Hence $\Gamma^{\mathcal{G}_{\beta\mathfrak{b}}}_\Sigma$ and $\Gamma^{\mathcal{G}_{\beta\eta\mathfrak{b}}}_\Sigma$ are acceptable.

[12] EdNote: needed Γ'_Σ saturated here

[13] EdNote: Chris: It seems to me that there is more needed: reflexivity and transitivity of $\doteq$ together with ∇_c. This derivation is not trivial to follow for the user and probably we should make it more explicit. As the same argument seems to apply to fdec, we may probably make it explicit just once here.

[14] EdNote: needed Γ'_Σ saturated

[15] EdNote: Chris: Couldn't we use here the same argument as in the soundness proof? If yes (and if this is coorect) we should use whatever is more easy to grap.

[16] EdNote: needed saturation

[17] EdNote: needed saturation

$\beta\xi$ For $\mathcal{G}_{\beta\xi}$, the rules $Init^{\dot{=}}$, dec^h (for all $h \in \Sigma$) and ξdec imply $\Gamma_\Sigma^{\mathcal{G}_{\beta\xi}}$ mates, logically decomposes and ξ-decomposes, hence $\Gamma_\Sigma^{\mathcal{G}_{\beta\xi}}$ is acceptable.

$\beta\xi\mathfrak{b}$ For $\mathcal{G}_{\beta\xi\mathfrak{b}}$, the rules $Init^{\dot{=}}$, dec^h (for parameters h) and ξdec imply $\Gamma_\Sigma^{\mathcal{G}_{\beta\xi\mathfrak{b}}}$ mates and ξ-decomposes, hence $\Gamma_\Sigma^{\mathcal{G}_{\beta\xi\mathfrak{b}}}$ is acceptable.

$\beta\mathfrak{f}$ For $\mathcal{G}_{\beta\mathfrak{f}}$, the rules $Init^{\dot{=}}$ and dec^h (for all $h \in \Sigma$) imply $\Gamma_\Sigma^{\mathcal{G}_{\beta\mathfrak{f}}}$ mates and logically decomposes, hence $\Gamma_\Sigma^{\mathcal{G}_{\beta\mathfrak{f}}}$ is acceptable.

$\beta\mathfrak{f}\mathfrak{b}$ For $\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}$, the rules $Init^{\dot{=}}$ and dec^h (for parameter h) imply $\Gamma_\Sigma^{\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}}$ mates and decomposes, hence $\Gamma_\Sigma^{\mathcal{G}_{\beta\mathfrak{f}\mathfrak{b}}}$ is acceptable.

So, each $\Gamma_\Sigma^{\mathcal{G}_*}$ is acceptable and sufficiently Σ-pure, and we can apply the saturation extension theorem 2.5.3 to obtain a saturated extension Γ'_Σ of $\Gamma_\Sigma^{\mathcal{G}_*}$.

Suppose we have $\mathcal{G}_*$-derivations of $\Gamma \to \mathbf{C}, \Delta$ and $\Gamma, \mathbf{C} \to \Delta$. By Lemma 2.3.13, $\Phi_{\Gamma\to\Delta} * \mathbf{C} \notin \Gamma'_\Sigma$ and $\Phi_{\Gamma\to\Delta} * \neg\mathbf{C} \notin \Gamma'_\Sigma$. By saturation of Γ'_Σ, we have $\Phi_{\Gamma\to\Delta} \notin \Gamma'_\Sigma$ and $\Phi_{\Gamma\to\Delta} \notin \Gamma_\Sigma^{\mathcal{G}_*}$ since $\Gamma_\Sigma^{\mathcal{G}_*} \subseteq \Gamma'_\Sigma$. Note that $\Phi_{\Gamma\to\Delta}$ must be sufficiently Σ-pure since it is finite. So, we must have a $\mathcal{G}_*$-derivation of $\Gamma \to \Delta$. $\Box$

As a corollary, by the discussion above, we know each $\Gamma_\Sigma^{\mathcal{G}_*}$ was already saturated.

Example 2.3.15 (Cut Is Not Always Admissible) It is worth noting that one can easily create a sequent calculus from the rules in Figures 2.1 and 2.2 which does *not* satisfy cut elimination. For example, consider a sequent calculus extending $\mathcal{G}_{\beta\eta}$ with rule b. This allows a limited incorporation of boolean extensionality, similar to what the higher order theorem prover TPS does. In this calculus, cut is not admissible. To see this, note that one can easily derive the sequents

$$a_o, b_o \to (a \doteq^o b)$$

and

$$(a \doteq^o b), (p_{o\to o}a) \to (pb)$$

but one cannot derive $a, b, (pa) \to (pb)$. It is also worth noting that if one also adds a cut rule to this system, the corresponding abstract consistency class in $\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$ is saturated. One can use this to show that we have (with the cut rule) a complete sequent calculus with respect to models in $\mathfrak{M}_{\beta\eta\mathfrak{b}}$. So, when creating a complete sequent calculus for extensionality, one has the choice of including a cut rule or including rules such as $Init^{\dot{=}}$ and decomposition. It seems clear that the latter choice is far better suited to automated theorem proving, as the cut rule is notoriously bad for automated proof search.

[18] As a consequence, we conclude that the landscape of notions of higher order semantics as defined in chapter 1 is very useful in the sense that for each model class in this framework we can indeed ensure respective cut elimination results. When developing a (machine-oriented) calculus for higher order logic we therefore advice the developer to first decide for a particular model class in this landscape and then to carefully design a (cut free but) complete calculus. EdNote(18) BegNP(19) EndNP(19)

Remark 2.3.16 (Resolution Calculi) One can also do a similar analysis for resolution calculi. In that case, showing saturation corresponds to showing that one can derive the empty clause from a set of clauses Φ whenever one can derive the empty clause from $\Phi * (\mathbf{A} \vee \neg\mathbf{A})$.

[18] EdNote: what does this mean, discuss the consequences!
[19] New Part: Chris: Can we say something as the following? I am not sure?

2.4 Model Existence without Saturation

2.4.1 Hintikka Sets

Hintikka sets connect syntax with semantics as they provide the basis for the model constructions in the model existence theorems. We define the Hintikka equivalents for the acceptability conditions: In the absence of saturation, we will need Hintikka sets to satisfy appropriate conditions to ensure the existence of models.

Definition 2.4.1 (Hintikka Acceptability Conditions) Let $\mathcal{H}$ be a Hintikka set in $\mathfrak{Hint}_*$. We define the following properties:

$\vec{\nabla}_m$ If $\mathbf{A}, \mathbf{B} \in cwff_o(\Sigma)$ are atomic and $\mathbf{A}, \neg\mathbf{B} \in \mathcal{H}$, then $\neg(\mathbf{A} \doteq^o \mathbf{B}) \in \mathcal{H}$.

$\vec{\nabla}^h_{dec}$ If $\neg(h\overline{\mathbf{A}^n} \doteq^\alpha h\overline{\mathbf{B}^n}) \in \mathcal{H}$ where $\alpha \in \{o, \iota\}$, then there is an i with $1 \leq i \leq n$ such that $\neg(\mathbf{A}^i \doteq \mathbf{B}^i) \in \mathcal{H}$.

$\vec{\nabla}_{\xi dec}$ If $h \in \Sigma$ and $\neg(h\overline{\mathbf{A}^n} \doteq^\alpha h\overline{\mathbf{B}^n}) \in \mathcal{H}$, then there is an i with $1 \leq i \leq n$ such that $\neg(\mathbf{A}^i \doteq \mathbf{B}^i) \in \mathcal{H}$.

$\vec{\nabla}_{fdec}$ If $\neg((\lambda\overline{X^m}.h\overline{\mathbf{A}^n}) \doteq^{\beta\to\gamma} (\lambda\overline{X^m}.h\overline{\mathbf{B}^n})) \in \mathcal{H}$ and $h \in \Sigma \cup \overline{x^m}$, then there is an i with $1 \leq i \leq n$ such that $\neg((\lambda\overline{X^m}.\mathbf{A}^i) \doteq (\lambda\overline{X^m}.\mathbf{B}^i)) \in \mathcal{H}$.

$\vec{\nabla}_k$ If $\neg(\lambda U_\alpha.\mathbf{A} \doteq \lambda U_\alpha.\mathbf{B}) \in \mathcal{H}$ where $U \notin free(\mathbf{A}) \cup free(\mathbf{B})$, then $\neg(\mathbf{A} \doteq \mathbf{B}) \in \mathcal{H}$.

We say $\mathcal{H}$ *mates* if it satisfies $\vec{\nabla}_m$. We say $\mathcal{H}$ *decomposes* if it satisfies $\vec{\nabla}^h_{dec}$ for every parameter $h \in \Sigma$. We say $\mathcal{H}$ *logically decomposes* if it satisfies $\vec{\nabla}^h_{dec}$ for every $h \in \Sigma$, including when h is $\neg$, $\lor$, or Π^α for some α. We say $\mathcal{H}$ *functionally decomposes* if it satisfies $\vec{\nabla}^h_{dec}$ for every parameter $h \in \Sigma$, $\vec{\nabla}_{fdec}$ and $\vec{\nabla}_k$. We say $\mathcal{H}$ *ξ-decomposes* if it satisfies $\vec{\nabla}^h_{dec}$ for every parameter $h \in \Sigma$ and $\vec{\nabla}_{\xi dec}$.

Definition 2.4.2 (Acceptable Hintikka Sets) We define when $\mathcal{H}$ is acceptable in $\mathfrak{Hint}_*$ as follows.

Every $\mathcal{H}$ in $\mathfrak{Hint}_{\beta\eta}$ [$\mathfrak{Hint}_\beta$] is *acceptable*.

A $\mathcal{H}$ in $\mathfrak{Hint}_{\beta\mathfrak{b}}$ [$\mathfrak{Hint}_{\beta\eta\mathfrak{b}}$] is *acceptable* if $\mathcal{H}$ mates and functionally decomposes.

A $\mathcal{H}$ in $\mathfrak{Hint}_{\beta\xi}$ is *acceptable* if $\mathcal{H}$ mates, logically decomposes and ξ-decomposes.

A $\mathcal{H}$ in $\mathfrak{Hint}_{\beta\xi\mathfrak{b}}$ is *acceptable* if $\mathcal{H}$ mates and ξ-decomposes.

A $\mathcal{H}$ in $\mathfrak{Hint}_{\beta\mathfrak{f}}$ is *acceptable* if $\mathcal{H}$ mates and logically decomposes.

A $\mathcal{H}$ in $\mathfrak{Hint}_{\beta\mathfrak{f}\mathfrak{b}}$ is *acceptable* if $\mathcal{H}$ mates and decomposes.

We will construct Hintikka sets as maximal elements of abstract consistency classes. To obtain a Hintikka set, we must explicitly show the property $\vec{\nabla}_\exists$ (and $\vec{\nabla}_\xi$ or $\vec{\nabla}_\mathfrak{f}$ when appropriate). This will ensure that Hintikka sets have parameters which act as witnesses.

Lemma 2.4.3 (Hintikka Lemma) *Let Γ_Σ be an abstract consistency class in $\mathfrak{Acc}_*$. Suppose a set $\mathcal{H} \in \Gamma_\Sigma$ satisfies the following properties:*

1. *$\mathcal{H}$ is maximal in Γ_Σ with respect to subset (i.e., for each sentence $\mathbf{D} \in cwff_o(\Sigma)$ such that $\mathcal{H} * \mathbf{D} \in \Gamma_\Sigma$, we already have $\mathbf{D} \in \mathcal{H}$).*

2. *$\mathcal{H}$ satisfies $\vec{\nabla}_\exists$.*

3. *If $* \in \{\beta\mathfrak{\xi}, \beta\mathfrak{\xi}\mathfrak{b}\}$, then $\vec{\nabla}_\mathfrak{\xi}$ holds in $\mathcal{H}$.*

4. *If $* \in \{\beta\mathfrak{f}, \beta\mathfrak{f}\mathfrak{b}\}$, then $\vec{\nabla}_\mathfrak{f}$ holds in $\mathcal{H}$.*

Then, $\mathcal{H} \in \mathfrak{Hint}_*$. If Γ_Σ is saturated, then $\mathcal{H}$ satisfies $\vec{\nabla}_{sat}$. If Γ_Σ is acceptable in $\mathfrak{Acc}_*$, then $\mathcal{H}$ is acceptable in $\mathfrak{Hint}_*$.

Proof: [20] $\mathcal{H}$ satisfies $\vec{\nabla}_\exists$ by assumption. Also, if $* \in \{\beta\mathfrak{f}, \beta\mathfrak{f}\mathfrak{b}\}$, then we have assumed $\mathcal{H}$ satisfies EdNote(20) $\vec{\nabla}_\mathfrak{f}$. The fact that $\mathcal{H} \in \Gamma_\Sigma$ satisfies $\vec{\nabla}_c$ follows directly from non-atomic consistency (Lemma 1.6.10). Every other $\vec{\nabla}_*$ property follows directly from the corresponding ∇_* property and maximality of $\mathcal{H}$ in Γ_Σ. For example, to show $\vec{\nabla}_\neg$, suppose $\neg\neg\mathbf{A} \in \mathcal{H}$. By $\nabla_\neg$, we know $\mathcal{H} * \mathbf{A} \in \Gamma_\Sigma$. By maximality of $\mathcal{H}$, we have $\mathbf{A} \in \mathcal{H}$. Checking $\vec{\nabla}_\beta$, $\vec{\nabla}_\eta$ (if $* \in \{\beta\eta, \beta\eta\mathfrak{b}\}$), $\vec{\nabla}_\wedge$, $\vec{\nabla}_\forall$, and $\vec{\nabla}_{\dot{=}}$ hold for $\mathcal{H}$ follows exactly this same pattern. Checking $\vec{\nabla}_\vee$, $\vec{\nabla}_\mathfrak{b}$ (if $* \in \{\beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\mathfrak{\xi}\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$) and $\vec{\nabla}_{sat}$ (if Γ_Σ is saturated) follows a similar pattern, but with a simple case analysis. For example, to check $\vec{\nabla}_{sat}$, given $\mathbf{A} \in cwff_o(\Sigma)$, ∇_{sat} implies $\mathcal{H} * \mathbf{A} \in \Gamma_\Sigma$ or $\mathcal{H} * \neg\mathbf{A} \in \Gamma_\Sigma$. So, either $\mathbf{A} \in \mathcal{H}$ or $\neg\mathbf{A} \in \mathcal{H}$. ☐

2.4.2 Abstract Extension Lemma

We shall now present the proof of the abstract extension lemma, which will nearly immediately yield the model existence theorems. For the proof we adapt the construction of Henkin's completeness proof from [Hen50, Hen96].[21] EdNote(21)

Theorem 2.4.4 (Abstract Extension Lemma) *Let Σ be a signature, Γ_Σ be a compact abstract consistency class in $\mathfrak{Acc}_*$ and let $\Phi \in \Gamma_\Sigma$ be sufficiently Σ-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. Also, if Γ_Σ is acceptable in $\mathfrak{Acc}_*$, then $\mathcal{H}$ is acceptable in $\mathfrak{Hint}_*$.*

In the following argument, note that α, β and γ are types as usual, while δ, ϵ, σ and τ are ordinals. **Proof:**[22] By Remark 1.3.16, there is an infinite cardinal $\aleph_s$ which is the cardinality of EdNote(22) Σ_α for each type α. This easily implies $cwff_\alpha(\Sigma)$ is of cardinality $\aleph_s$ for each type α. Let ϵ be the first ordinal of this cardinality. (In the countable case, ϵ is ω.) Since the cardinality of $cwff_o(\Sigma)$ is $\aleph_s$, we can use the well-ordering principle to enumerate $cwff_o(\Sigma)$ as $(\mathbf{A}^\delta)_{\delta<\epsilon}$.

Let α be a type. For each $\delta < \epsilon$, let U_α^δ be the set of constants which occur in a sentence in the set $\{\mathbf{A}^\sigma \mid \sigma \leq \delta\}$. Since $\delta < \epsilon$, the set $\{\mathbf{A}^\sigma \mid \sigma \leq \delta\}$ has cardinality less than $\aleph_s$. Hence, U_α^δ has cardinality less than $\aleph_s$. By sufficient purity, we know there is a set of parameters $P_\alpha \subseteq \Sigma_\alpha$ of cardinality $\aleph_s$ such that the parameters in P_α do not occur in Φ. Since, considering cardinality, we cannot have $P_\alpha \subseteq U_\alpha^\delta$ for any $\delta < \epsilon$, we know $P_\alpha \setminus U_\alpha^\delta$ is non-empty for each $\delta < \epsilon$. Using the axiom of choice, we can find a sequence $(w_\alpha^\delta)_{\delta<\epsilon}$ where for each $\delta < \epsilon$, $w_\alpha^\delta \in P_\alpha \setminus U_\alpha^\delta$. That is, for each type α, we know w_α^δ is a parameter of type α which does not occur in $\Phi \cup \{\mathbf{A}^\sigma \mid \sigma \leq \delta\}$. As a consequence, if w_α^δ occurs in $\mathbf{A}^\sigma$, then $\delta < \sigma$.

[20] EdNote: update proof to include acceptable

[21] EdNote: the proof of the following theorem should be reduced to the new parts

[22] EdNote: update proof to include acceptability

The parameters w^δ_α are intended to serve as witnesses. To ease the argument, we define two sequences of witnessing sentences related to the sequence $(\mathbf{A}^\delta)_{\delta<\epsilon}$. For each $\delta < \epsilon$, let $\mathbf{E}^\delta := \neg(\mathbf{B}w^\delta_\alpha)$ if $\mathbf{A}^\delta$ is of the form $\neg(\Pi^\alpha\mathbf{B})$, and let $\mathbf{E}^\delta := \mathbf{A}^\delta$ otherwise. If $* \in \{\beta\mathfrak{f}, \beta\mathfrak{fb}\}$ and $\mathbf{A}^\delta$ is of the form $\neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G})$, let $\mathbf{X}^\delta := \neg(\mathbf{F}w^\delta_\alpha \doteq^\beta \mathbf{G}w^\delta_\alpha)$. If $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$ and $\mathbf{A}^\delta$ is of the form $\neg((\lambda X_\alpha.\mathbf{M}) \doteq^{\alpha\to\beta} (\lambda X.\mathbf{N}))$, let $\mathbf{X}^\delta := \neg([w^\delta_\alpha/X]\mathbf{M} \doteq^\beta [w^\delta_\alpha/X]\mathbf{N})$. Otherwise, let $\mathbf{X}^\delta := \mathbf{A}^\delta$. (Notice that any sentence $\neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G})$ is also of the form $\neg(\Pi^\gamma\mathbf{B})$, where γ is $(\alpha \to \beta) \to o$. So, whenever $\mathbf{X}^\delta \not\equiv \mathbf{A}^\delta$, we must also have $\mathbf{E}^\delta \not\equiv \mathbf{A}^\delta$.)

We construct $\mathcal{H}$ by inductively constructing a transfinite sequence $(\mathcal{H}^\delta)_{\delta<\epsilon}$ such that $\mathcal{H}^\delta \in \Gamma_\Sigma$ for each $\delta < \epsilon$. Then the Σ-Hintikka set is $\mathcal{H} := \bigcup_{\delta<\epsilon}\mathcal{H}^\delta$. We define $\mathcal{H}^0 := \Phi$. For limit ordinals δ, we define $\mathcal{H}^\delta := \bigcup_{\sigma<\delta}\mathcal{H}^\sigma$.

In the successor case, if $\mathcal{H}^\delta * \mathbf{A}^\delta \in \Gamma_\Sigma$, then we let $\mathcal{H}^{\delta+1} := \mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta * \mathbf{X}^\delta$. If $\mathcal{H}^\delta * \mathbf{A}^\delta \notin \Gamma_\Sigma$, we let $\mathcal{H}^{\delta+1} := \mathcal{H}^\delta$.

We show by induction that for every $\delta < \epsilon$, type α and parameter w^τ_α which occurs in some sentence in $\mathcal{H}^\delta$, we have $\tau < \delta$. The base case holds since no w^τ_α occurs in any sentence in $\mathcal{H}^0 \equiv \Phi$. For any limit ordinal δ, if w^τ_α occurs in some sentence in $\mathcal{H}^\delta$, then by definition of $\mathcal{H}^\delta$, w^τ_α already occurs in some sentence in $\mathcal{H}^\sigma$ for some $\sigma < \delta$. So, $\tau < \sigma < \delta$.

For any successor ordinal $\delta + 1$, suppose w^τ_α occurs in some sentence in $\mathcal{H}^{\delta+1}$. If it already occurred in a sentence in $\mathcal{H}^\delta$, then we have $\tau < \delta < \delta + 1$ by the inductive assumption. So, we need only consider the case where w^τ_α occurs in a sentence in $\mathcal{H}^{\delta+1} \setminus \mathcal{H}^\delta$. Note that $(\mathcal{H}^{\delta+1} \setminus \mathcal{H}^\delta) \subseteq \{\mathbf{A}^\delta, \mathbf{E}^\delta, \mathbf{X}^\delta\}$. In any case, note that if τ is δ, then we are done, since $\delta < \delta + 1$. If w^τ_α is any parameter with $\tau \not\equiv \delta$ and occurs in $\mathbf{E}^\delta$ or $\mathbf{X}^\delta$, then it must also occur in $\mathbf{A}^\delta$ (by inspecting the possible definitions of $\mathbf{E}^\delta$ and $\mathbf{X}^\delta$), in which case $\tau < \delta < \delta + 1$.

In particular, we now know w^δ_α does not occur in any sentence in $\mathcal{H}^\delta$ for any $\delta < \epsilon$ and type α.

Next we show by induction that $\mathcal{H}^\delta \in \Gamma_\Sigma$ for all $\delta < \epsilon$. The base case holds by the assumption that $\mathcal{H}^0 \equiv \Phi \in \Gamma_\Sigma$. For any limit ordinal δ, assume $\mathcal{H}^\sigma \in \Gamma_\Sigma$ for every $\sigma < \delta$. We have $\mathcal{H}^\delta \equiv \bigcup_{\sigma<\delta}\mathcal{H}^\sigma \in \Gamma_\Sigma$ by compactness, since any finite subset of $\mathcal{H}^\delta$ is a subset of $\mathcal{H}^\sigma$ for some $\sigma < \delta$.

For any successor ordinal $\delta + 1$, we assume $\mathcal{H}^\delta \in \Gamma_\Sigma$. We have to show that $\mathcal{H}^{\delta+1} \in \Gamma_\Sigma$. This is trivial in case $\mathcal{H}^\delta * \mathbf{A}^\delta \notin \Gamma_\Sigma$ (for all abstract consistency classes) since $\mathcal{H}^{\delta+1} \equiv \mathcal{H}^\delta$. Suppose $\mathcal{H}^\delta * \mathbf{A}^\delta \in \Gamma_\Sigma$. We consider three sub-cases:

1. If $\mathbf{E}^\delta \equiv \mathbf{A}^\delta$ and $\mathbf{X}^\delta \equiv \mathbf{A}^\delta$, then $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta * \mathbf{X}^\delta \in \Gamma_\Sigma$ since $\mathcal{H}^\delta * \mathbf{A}^\delta \in \Gamma_\Sigma$.

2. If $\mathbf{E}^\delta \not\equiv \mathbf{A}^\delta$ and $\mathbf{X}^\delta \equiv \mathbf{A}^\delta$, then $\mathbf{A}^\delta$ is of the form $\neg\Pi^\alpha\mathbf{B}$ and $\mathbf{E}^\delta \equiv \neg\mathbf{B}w^\delta_\alpha$. We conclude that $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta \in \Gamma_\Sigma$ by $\nabla_\exists$ since w^δ_α does not occur in $\mathbf{A}^\delta$ or any sentence in $\mathcal{H}^\delta$. Since $\mathbf{X}^\delta \equiv \mathbf{A}^\delta$, this is the same as concluding $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta * \mathbf{X}^\delta \in \Gamma_\Sigma$.

3. If $\mathbf{X}^\delta \not\equiv \mathbf{A}^\delta$, then $* \in \{\beta\xi, \beta\mathfrak{f}, \beta\xi\mathfrak{b}, \beta\mathfrak{fb}\}$ (by the definition of $\mathbf{X}^\delta$). $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta \in \Gamma_\Sigma$ by $\nabla_\exists$ since $w^\delta_{(\alpha\to\beta)\to o}$ does not occur in $\mathbf{A}^\delta$ or any sentence in $\mathcal{H}^\delta$. Now, w^δ_α (which is different from $w^\delta_{(\alpha\to\beta)\to o}$ since it has a different type) does not occur in any sentence in $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta$. We have $\mathcal{H}^\delta * \mathbf{A}^\delta * \mathbf{E}^\delta * \mathbf{X}^\delta \in \mathcal{H}$ by ∇_ξ (if $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$) or by $\nabla_\mathfrak{f}$ (if $* \in \{\beta\mathfrak{f}, \beta\mathfrak{fb}\}$).

Since Γ_Σ is compact, we also have $\mathcal{H} \in \Gamma_\Sigma$.

Now we know that our inductively defined set $\mathcal{H}$ is indeed in Γ_Σ and that $\Phi \subseteq \mathcal{H}$. In order to apply Lemma 1.6.22, we must check $\mathcal{H}$ is maximal, satisfies $\vec{\nabla}_\exists$, $\vec{\nabla}_\xi$ (if $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$), and $\vec{\nabla}_\mathfrak{f}$ (if $* \in \{\beta\mathfrak{f}, \beta\mathfrak{fb}\}$). It is immediate from the construction that $\vec{\nabla}_\exists$ holds since if $\neg(\Pi^\alpha\mathbf{F}) \in \mathcal{H}$, then $\neg(\mathbf{F}w^\delta_\alpha) \in \mathcal{H}$ where δ is the ordinal such that $\mathbf{A}^\delta \equiv \neg(\Pi^\alpha\mathbf{F})$. If $* \in \{\beta\xi, \beta\xi\mathfrak{b}\}$, then we have ensured $\vec{\nabla}_\xi$ holds since $\neg([w^\delta_\alpha/X]\mathbf{M} \doteq^\beta [w^\delta_\alpha/X]\mathbf{N}) \in \mathcal{H}$ whenever $\neg((\lambda X_\alpha.\mathbf{M}) \doteq^{\alpha\to\beta} (\lambda X.\mathbf{N})) \in \mathcal{H}$

where δ is the ordinal such that $\mathbf{A}^\delta \equiv \neg((\lambda X_\alpha.\mathbf{M}) \doteq^{\alpha\to\beta} (\lambda X.\mathbf{N}))$. Similarly, we have ensured $\vec{\nabla}_\mathfrak{f}$ holds when $* \in \{\beta\mathfrak{f}, \beta\mathfrak{fb}\}$ since $\neg(\mathbf{F}w^\delta_\alpha \doteq^\beta \mathbf{G}w^\delta_\alpha) \in \mathcal{H}$ whenever $\neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G}) \in \mathcal{H}$ where δ is the ordinal such that $\mathbf{A}^\delta \equiv \neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G})$.

It only remains to show that $\mathcal{H}$ is maximal in Γ_Σ. So, let $\mathbf{A} \in cwff_o(\Sigma)$ and $\mathcal{H} * \mathbf{A} \in \Gamma_\Sigma$ be given. Note that $\mathbf{A} \equiv \mathbf{A}^\delta$ for some $\delta < \epsilon$. Since $\mathcal{H}$ is closed under subsets we know that $\mathcal{H}^\delta * \mathbf{A}^\delta \in \Gamma_\Sigma$. By definition of $\mathcal{H}^{\delta+1}$ we conclude that $\mathbf{A}^\delta \in \mathcal{H}^{\delta+1}$ and hence $\mathbf{A} \in \mathcal{H}$.

So, Lemma 1.6.22 implies $\mathcal{H} \in \mathfrak{Hint}_*$ and $\mathcal{H}$ is saturated if Γ_Σ is saturated. □

[23] EdNote(23)

2.4.3 Constructing Models from Partial Equivalence Relations

When constructing models below, we will make use of quotients obtained from pers (partial equivalence relations, i.e., symmetric, transitive relations). This is more general than taking quotients by a total equivalence relations, but we must require more conditions for the quotient to be well-defined. First, we develop the theory of such constructions.

Definition 2.4.5 (Σ-Per Evaluation) A Σ-*per evaluation* is a triple $(\mathcal{A}, \sim, \upsilon)$ where $\mathcal{A} \equiv (\mathcal{D}, @, \mathcal{E})$ is a Σ-evaluation, $\upsilon : \mathcal{D}_o \to \{\mathtt{T}, \mathtt{F}\}$, and $\sim$ be a typed per on the domains $\mathcal{D}_\alpha$. At each type α, we define $\overline{\mathcal{D}_\alpha^\sim} := \{\mathsf{a} \in \mathcal{D}_\alpha | \mathsf{a} \sim \mathsf{a}\}$. Furthermore, for any two assignments φ and ψ, we use the notation $\varphi \sim \psi$ to indicate that $\varphi(x) \sim \psi(x)$ for every variable x.

Definition 2.4.6 (Σ-Per Evaluation Properties) Let $(\mathcal{A}, \sim, \upsilon)$ be a Σ-per evaluation. We define the following properties for a per evaluation.

$\partial^{o,\iota}$: $\sim$ is a total equivalence relation on $\mathcal{D}_\iota$ and $\mathcal{D}_o$.

∂^Σ: For every constant $c_\alpha \in \Sigma$, $\mathcal{E}(c) \sim \mathcal{E}(c)$ in $\mathcal{D}_\alpha$.

∂^c: For every $\mathsf{g}, \mathsf{h} \in \mathcal{D}_{\alpha\to\beta}$ and $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$, if $\mathsf{g} \sim \mathsf{h}$ and $\mathsf{a} \sim \mathsf{b}$, then $\mathsf{g}@\mathsf{a} \sim \mathsf{h}@\mathsf{b}$.

∂^s: For every type α, every $\mathbf{A} \in wff_\alpha(\Sigma)$, and all assignments φ and ψ, if $\varphi \sim \psi$, then $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\psi(\mathbf{A})$.

∂^υ: For all $\mathsf{a}, \mathsf{b} \in \mathcal{D}_o$, $\mathsf{a} \sim \mathsf{b}$ implies $\upsilon(\mathsf{a}) \equiv \upsilon(\mathsf{b})$.

$\partial^\neg$: For all $\mathsf{a} \in \overline{\mathcal{D}_o^\sim}$, $\upsilon(\mathcal{E}(\neg)@\mathsf{a}) \equiv \mathtt{T}$ iff $\upsilon(\mathsf{a}) \equiv \mathtt{F}$.

$\partial^\vee$: For all $\mathsf{a}, \mathsf{b} \in \overline{\mathcal{D}_o^\sim}$, $\upsilon(\mathcal{E}(\vee)@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$ iff $\upsilon(\mathsf{a}) \equiv \mathtt{T}$ or $\upsilon(\mathsf{b}) \equiv \mathtt{T}$.

∂^Π: For all $\mathsf{f} \in \overline{\mathcal{D}_{\alpha\to o}^\sim}$, $\upsilon(\mathcal{E}(\Pi^\alpha)@\mathsf{f}) \equiv \mathtt{T}$ iff $\upsilon(\mathsf{f}@\mathsf{a}) \equiv \mathtt{T}$ for each $\mathsf{a} \in \overline{\mathcal{D}_\alpha^\sim}$.

$\partial^\mathfrak{q}$: At each type $\alpha \in \mathcal{T}$, there is an element $\mathrm{q}^\alpha \in \overline{\mathcal{D}_{\alpha\to\alpha\to o}^\sim}$, such that for all $\mathsf{a}, \mathsf{b} \in \overline{\mathcal{D}_\alpha^\sim}$ we have $\upsilon(\mathrm{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$ iff $\mathsf{a} \sim \mathsf{b}$.

∂^η: For every type α, $\mathbf{A} \in wff_\alpha(\Sigma)$, and assignment φ, $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\varphi(\mathbf{A}{\downarrow}_{\beta\eta})$.

∂^ξ: For all $\alpha, \beta \in \mathcal{T}$, $\mathbf{M}, \mathbf{N} \in wff_\beta(\Sigma)$, assignments φ with $\varphi \sim \varphi$, and variables X_α, $\mathcal{E}_\varphi(\lambda X_\alpha.\mathbf{M}_\beta) \sim \mathcal{E}_\varphi(\lambda X_\alpha.\mathbf{N}_\beta)$ whenever $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \sim \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N})$ for every $\mathsf{a} \in \overline{\mathcal{D}_\alpha^\sim}$.

[23] EDNOTE: MiKo: I have deleted the section on "Model Existence with saturation". That is already in the first chapter.

∂^{f}**:** For every $\mathsf{g}, \mathsf{h} \in \mathcal{D}_{\alpha\to\beta}$, $\mathsf{g} \sim \mathsf{h}$ whenever for every $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ $\mathsf{a} \sim \mathsf{b}$ implies $\mathsf{g}@\mathsf{a} \sim \mathsf{h}@\mathsf{b}$.

∂^{b}**:** There are only two $\sim$-equivalence classes on $\mathcal{D}_o$.

When constructing a per evaluation satisfying ∂^c and ∂^{f}, it is enough to specify $\sim$ only on the base types, since this determines $\sim$ on all function types. We will formulate this in terms of applicative structures, so we can build such pers before we interpret terms.

Definition 2.4.7 (Functional Per Extension) Let $(\mathcal{D}, @)$ be an applicative structure, $\sim_o$ be a partial equivalence relation on $\mathcal{D}_o$, and $\sim_\iota$ be a partial equivalence relation on $\mathcal{D}_\iota$. We define the *functional per extension* $\sim$ of $\sim_o$ and $\sim_\iota$ on $(\mathcal{D}, @)$ by induction on types.

o**:** For $\mathsf{a}, \mathsf{b} \in \mathcal{D}_o$, $\mathsf{a} \sim \mathsf{b}$ if $\mathsf{a} \sim_o \mathsf{b}$.

ι**:** For $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\iota$, $\mathsf{a} \sim \mathsf{b}$ if $\mathsf{a} \sim_\iota \mathsf{b}$.

$\beta \to \gamma$**:** For $\mathsf{g}, \mathsf{h} \in \mathcal{D}_{\beta\to\gamma}$, $\mathsf{g} \sim \mathsf{h}$ if for every $\mathsf{a} \sim \mathsf{b}$ in $\mathcal{D}_\beta$ we have $\mathsf{g}@\mathsf{a} \sim \mathsf{h}@\mathsf{b}$.

Note that $\sim$ agrees with $\sim_o$ on $\mathcal{D}_o$ and $\sim_\iota$ on $\mathcal{D}_\iota$.

It is immediate by the definition at function types that ∂^c and ∂^{f} will hold in a per evaluation $((\mathcal{D}, @, \mathcal{E}), \sim, \upsilon)$ if $\sim$ is the functional per extension of any pers on $\mathcal{D}_o$ and $\mathcal{D}_\iota$.

Lemma 2.4.8 *The functional per extension $\sim$ of $\sim_o$ and $\sim_\iota$ on $(\mathcal{D}, @)$ is a typed per on the domains $\mathcal{D}_\alpha$.*

Proof: We need to show $\sim$ is symmetric and transitive on each $\mathcal{D}_\alpha$. We can prove this by induction on α.

$\alpha \in \{o, \iota\}$ This is immediate since $\sim$ agrees with $\sim_\alpha$ which was assumed to be a per.

$\alpha \equiv (\beta \to \gamma)$**:** Suppose $\mathsf{g} \sim \mathsf{h}$ in $\mathcal{D}_{\beta\to\gamma}$. Let $\mathsf{a} \sim \mathsf{b}$ in $\mathcal{D}_\beta$ be given. By induction, $\sim$ is symmetric on $\mathcal{D}_\beta$, so $\mathsf{b} \sim \mathsf{a}$. By definition, $\mathsf{g}@\mathsf{b} \sim \mathsf{h}@\mathsf{a}$. By induction, $\sim$ is symmetric on $\mathcal{D}_\gamma$, so $\mathsf{h}@\mathsf{a} \sim \mathsf{g}@\mathsf{b}$. Generalizing on a and b, we have $\mathsf{h} \sim \mathsf{g}$, showing symmetry.

For transitivity, suppose $\mathsf{g} \sim \mathsf{h}$ and $\mathsf{h} \sim \mathsf{k}$ in $\mathcal{D}_{\beta\to\gamma}$. Let $\mathsf{a} \sim \mathsf{b}$ in $\mathcal{D}_\beta$ be given. By induction, $\sim$ is symmetric and transitive on $\mathcal{D}_\beta$, so $\mathsf{a} \sim \mathsf{b} \sim \mathsf{a}$ and $\mathsf{a} \sim \mathsf{a}$. Thus, $\mathsf{g}@\mathsf{a} \sim \mathsf{h}@\mathsf{a} \sim \mathsf{k}@\mathsf{b}$. By induction, $\sim$ is transitive on $\mathcal{D}_\gamma$, so $\mathsf{g}@\mathsf{a} \sim \mathsf{k}@\mathsf{b}$. Generalizing on a and b, we have $\mathsf{g} \sim \mathsf{k}$, showing transitivity.

$\square$

Lemma 2.4.9 *Let $\mathcal{P} \equiv (\mathcal{A}, \sim, \upsilon)$ be a Σ-per evaluation with $\mathcal{A} \equiv (\mathcal{D}, @, \mathcal{E})$. Suppose $\mathcal{P}$ satisfies ∂^Σ, ∂^c and ∂^{f}. Then $\mathcal{P}$ satisfies ∂^s.*

Proof: We prove $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\psi(\mathbf{A})$ for all assignments φ and ψ and every $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$ by induction on $\mathbf{A}$. For constants, this follows from ∂^Σ. For a variable x_α, this follows from $\mathcal{E}_\varphi(x) \equiv \varphi(x) \sim \psi(x) \equiv \mathcal{E}_\psi(x)$. For application, we assume $\mathcal{E}_\varphi(\mathbf{G}) \sim \mathcal{E}_\psi(\mathbf{G})$ and $\mathcal{E}_\varphi(\mathbf{C}) \sim \mathcal{E}_\psi(\mathbf{C})$. By ∂^c, we know

$$\mathcal{E}_\varphi(\mathbf{GC}) \equiv \mathcal{E}_\varphi(\mathbf{G})@\mathcal{E}_\varphi(\mathbf{C}) \sim \mathcal{E}_\psi(\mathbf{G})@\mathcal{E}_\psi(\mathbf{C}) \equiv \mathcal{E}_\psi(\mathbf{GC}).$$

For abstraction we use ∂^{f}. We need to show $\mathcal{E}_\varphi(\lambda X_\alpha.\mathbf{C}_\gamma)$ and $\mathcal{E}_\psi(\lambda X_\alpha.\mathbf{C}_\gamma)$ send related values to related results. Let $\mathsf{a} \sim \mathsf{b}$ in $\mathcal{D}_\alpha$ be given. By induction, we have $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{C}) \sim \mathcal{E}_{\psi,[\mathsf{b}/X]}(\mathbf{C})$, so

$$\mathcal{E}_\varphi(\lambda X.\mathbf{C})@\mathsf{a} \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{C}) \sim \mathcal{E}_{\psi,[\mathsf{b}/X]}(\mathbf{C}) \equiv \mathcal{E}_\psi(\lambda X.\mathbf{C})@\mathsf{b}$$

By ∂^{f}, $\mathcal{E}_\varphi(\lambda X.\mathbf{C}) \sim \mathcal{E}_\psi(\lambda X.\mathbf{C})$. $\square$

Theorem 2.4.10 *Let $\mathcal{P} \equiv (\mathcal{A}, \sim, \upsilon)$ be a Σ-per evaluation with $\mathcal{A} \equiv (\mathcal{D}, @, \mathcal{E})$. Suppose $\mathcal{P}$ satisfies $\partial^{o,\iota}$, ∂^{Σ}, ∂^{c}, ∂^{s}, ∂^{v}, $\partial^{\neg}$, $\partial^{\vee}$ and ∂^{Π}. Then, there is a Σ-model $\mathcal{M} \equiv (\mathcal{D}^{\sim}, @^{\sim}, \mathcal{E}^{\sim}, \upsilon^{\sim})$ such that $\upsilon^{\sim}(\mathcal{E}^{\sim}(\mathbf{A})) \equiv \upsilon(\mathcal{E}(\mathbf{A}))$ for all $\mathbf{A} \in cwff_o(\Sigma)$. Furthermore, we have:*

1. *If $\mathcal{P}$ satisfies $\partial^{\mathfrak{q}}$, then $\mathcal{M}$ satisfies property $\mathfrak{q}$.*
2. *If $\mathcal{P}$ satisfies ∂^{η}, then $\mathcal{M}$ satisfies property η.*
3. *If $\mathcal{P}$ satisfies ∂^{ξ}, then $\mathcal{M}$ satisfies property ξ.*
4. *If $\mathcal{P}$ satisfies $\partial^{\mathfrak{f}}$, then $\mathcal{M}$ satisfies property $\mathfrak{f}$.*
5. *If $\mathcal{P}$ satisfies $\partial^{\mathfrak{b}}$, then $\mathcal{M}$ satisfies property $\mathfrak{b}$.*

Proof: We define the domains $\mathcal{D}^{\sim}$ of $\mathcal{M}$ as $\mathcal{D}_{\alpha}^{\sim} := \{[\![\mathsf{a}]\!]_{\sim} | \mathsf{a} \in \overline{\mathcal{D}_{\alpha}^{\sim}}\}$. It is helpful to choose representatives of the equivalence classes $\mathsf{A} \in \mathcal{D}_{\alpha}^{\sim}$ as in Definition 1.3.33. We use the axiom of choice to obtain such a (total) function from $\mathcal{D}_{\alpha}^{\sim}$ to $\overline{\mathcal{D}_{\alpha}^{\sim}}$. We will denote this operation by A^*. So, for each $\mathsf{A} \in \mathcal{D}_{\alpha}^{\sim}$,

$$\mathsf{A}^* \in \overline{\mathcal{D}_{\alpha}^{\sim}} \quad \text{and} \quad \mathsf{A} \equiv [\![\mathsf{A}^*]\!]_{\sim}$$

It follows that if $\mathsf{A}, \mathsf{B} \in \mathcal{D}_{\alpha}^{\sim}$ and $\mathsf{A}^* \sim \mathsf{B}^*$, then

$$\mathsf{A} \equiv [\![\mathsf{A}^*]\!]_{\sim} \equiv [\![\mathsf{B}^*]\!]_{\sim} \equiv \mathsf{B}$$

Also, for any $\mathsf{a} \in \overline{\mathcal{D}_{\alpha}^{\sim}}$, $\mathsf{a} \in [\![\mathsf{a}]\!]_{\sim}$ and $[\![\mathsf{a}]\!]_{\sim}^* \in [\![\mathsf{a}]\!]_{\sim}$, so we have

$$\mathsf{a} \sim [\![\mathsf{a}]\!]_{\sim}^*$$

For each assignment φ taking variables x_{α} to $\mathcal{D}_{\alpha}^{\sim}$, we can define φ^* to be an assignment taking variables x_{α} to $\overline{\mathcal{D}_{\alpha}^{\sim}}$ by $\varphi^*(x) := \varphi(x)^*$.

We define $@^{\sim}$ by

$$\mathsf{G} @^{\sim} \mathsf{A} := [\![\mathsf{G}^* @ \mathsf{A}^*]\!]_{\sim}$$

for $\mathsf{G} \in \mathcal{D}_{\alpha\to\beta}^{\sim}$ and $\mathsf{A} \in \mathcal{D}_{\alpha}^{\sim}$. Note that for any $\mathsf{g} \in \overline{\mathcal{D}_{\alpha\to\beta}^{\sim}}$ and $\mathsf{a} \in \overline{\mathcal{D}_{\alpha}^{\sim}}$, we have $[\![\mathsf{g}]\!]_{\sim}^* @ [\![\mathsf{a}]\!]_{\sim}^* \sim \mathsf{g} @ \mathsf{a}$ by ∂^{c}, $\mathsf{g} \sim [\![\mathsf{g}]\!]_{\sim}^*$ and $\mathsf{a} \sim [\![\mathsf{a}]\!]_{\sim}^*$. As a result, we have

$$[\![\mathsf{g}]\!]_{\sim} @^{\sim} [\![\mathsf{a}]\!]_{\sim} \equiv [\![\mathsf{g} @ \mathsf{a}]\!]_{\sim}$$

We next define $\mathcal{E}_{\varphi}^{\sim}(\mathbf{A}) := [\![\mathcal{E}_{\varphi^*}(\mathbf{A})]\!]_{\sim}$. To check this is well-defined, we need to know $\mathcal{E}_{\varphi^*}(\mathbf{A}_{\alpha}) \in \overline{\mathcal{D}_{\alpha}^{\sim}}$ for each $\mathbf{A} \in wff_{\alpha}(\Sigma)$. This follows directly from ∂^{s} since $\varphi^*(x) \sim \varphi^*(x)$ for every variable x.

We check $\mathcal{E}^{\sim}$ is an evaluation function.

1. For each variable x_{α}, $\mathcal{E}_{\varphi}^{\sim}(x) \equiv [\![\varphi^*(x)]\!]_{\sim} \equiv \varphi(x)$.
2. $\mathcal{E}^{\sim}$ preserves application since

   $$\mathcal{E}_{\varphi}^{\sim}(\mathbf{GC}) \equiv [\![\mathcal{E}_{\varphi^*}(\mathbf{GC})]\!]_{\sim} \equiv [\![\mathcal{E}_{\varphi^*}(\mathbf{G}) @ \mathcal{E}_{\varphi^*}(\mathbf{C})]\!]_{\sim} \equiv [\![\mathcal{E}_{\varphi^*}(\mathbf{G})]\!]_{\sim} @^{\sim} [\![\mathcal{E}_{\varphi^*}(\mathbf{C})]\!]_{\sim} \equiv \mathcal{E}_{\varphi}^{\sim}(\mathbf{G}) @^{\sim} \mathcal{E}_{\varphi}^{\sim}(\mathbf{C})$$

3. If φ and ψ coincide on the free variables of $\mathbf{A}$, then so do φ^* and ψ^*. So, $\mathcal{E}_{\varphi^*}(\mathbf{A}) \equiv \mathcal{E}_{\psi^*}(\mathbf{A})$ since $\mathcal{A}$ is a Σ-evaluation. This directly implies $\mathcal{E}_{\varphi}^{\sim}(\mathbf{A}) \equiv \mathcal{E}_{\psi}^{\sim}(\mathbf{A})$.
4. Let φ be an assignment and $\mathbf{A} \in wff_{\alpha}(\Sigma)$. Since $\mathcal{A}$ is a Σ-evaluation, we have $\mathcal{E}_{\varphi^*}(\mathbf{A}) \equiv \mathcal{E}_{\varphi^*}(\mathbf{A}{\downarrow_{\beta}})$. Since $\mathcal{E}_{\varphi^*}(\mathbf{A}) \in \overline{\mathcal{D}_{\alpha}^{\sim}}$, we can pass to equivalence classes and obtain $\mathcal{E}_{\varphi}^{\sim}(\mathbf{A}) \equiv \mathcal{E}_{\varphi}^{\sim}(\mathbf{A}{\downarrow_{\beta}})$ as desired.

So, $\mathcal{E}^{\sim}$ is an evaluation function.

We define $\upsilon^{\sim} : \mathcal{D}^{\sim}_{o} \to \{\mathtt{T}, \mathtt{F}\}$ by

$$\upsilon^{\sim}(\mathsf{A}) := \upsilon(\mathsf{A}^{*})$$

for $\mathsf{A} \in \mathcal{D}^{\sim}_{o}$. For each $\mathsf{a} \in \overline{\mathcal{D}^{\sim}_{o}}$, from ∂^{υ} we conclude

$$\upsilon^{\sim}([\![\mathsf{a}]\!]_{\sim}) \equiv \upsilon([\![\mathsf{a}]\!]^{*}_{\sim}) \equiv \upsilon(\mathsf{a})$$

By $\partial^{o,\iota}$, $\mathsf{a} \sim \mathsf{a}$ and $\mathsf{b} \sim \mathsf{b}$, so $\upsilon^{\sim}([\![\mathsf{a}]\!]_{\sim}) \equiv \mathtt{T}$ and $\upsilon^{\sim}([\![\mathsf{b}]\!]_{\sim}) \equiv \mathtt{F}$.

An important property of $\upsilon^{\sim}$ is that if $\mathsf{A} \in \mathcal{D}^{\sim}_{\alpha}$ and $\mathsf{a} \in \mathsf{A}$, then $\upsilon^{\sim}(\mathsf{A}) \equiv \upsilon(\mathsf{a})$. This follows from ∂^{υ}, since $\mathsf{A}^{*} \sim \mathsf{a}$. This will be used several times to verify the required properties of $\upsilon^{\sim}$ below. In particular, $\mathcal{E}(\mathbf{A}) \in \overline{\mathcal{D}^{\sim}_{o}}$ implies $\mathcal{E}(\mathbf{A}) \in \mathcal{E}^{\sim}(\mathbf{A})$ and so

$$\upsilon^{\sim}(\mathcal{E}^{\sim}(\mathbf{A})) \equiv \upsilon(\mathcal{E}(\mathbf{A}))$$

as required. We finally check that $\mathcal{M}$ is a model by checking the conditions on $\upsilon^{\sim}$.

1. $\upsilon^{\sim}(\mathcal{E}^{\sim}(\neg)@^{\sim}\mathsf{A}) \equiv \mathtt{T}$, iff $\upsilon(\mathcal{E}(\neg)@\mathsf{A}^{*}) \equiv \mathtt{T}$, iff (by $\partial^{\neg}$) $\upsilon(\mathsf{A}^{*}) \equiv \mathtt{F}$, iff $\upsilon^{\sim}(\mathsf{A}) \equiv \mathtt{F}$
2. $\upsilon^{\sim}(\mathcal{E}^{\sim}(\vee)@^{\sim}\mathsf{A}@^{\sim}\mathsf{B}) \equiv \mathtt{T}$, iff $\upsilon(\mathcal{E}(\vee)@\mathsf{A}^{*}@\mathsf{B}^{*}) \equiv \mathtt{T}$, iff (by $\partial^{\vee}$) $\upsilon(\mathsf{A}^{*}) \equiv \mathtt{T}$ or $\upsilon(\mathsf{B}^{*}) \equiv \mathtt{T}$, iff $\upsilon^{\sim}(\mathsf{A}) \equiv \mathtt{T}$ or $\upsilon^{\sim}(\mathsf{B}) \equiv \mathtt{T}$,
3. Let $\mathsf{P} \in \mathcal{D}^{\sim}_{\alpha\to o}$ be given. $\upsilon^{\sim}(\mathcal{E}^{\sim}(\Pi^{\alpha})@^{\sim}\mathsf{P}) \equiv \mathtt{T}$, iff $\upsilon(\mathcal{E}(\Pi^{\alpha})@\mathsf{P}^{*}) \equiv \mathtt{T}$, iff (by ∂^{Π}) $\upsilon(\mathsf{P}^{*}@\mathsf{a}) \equiv \mathtt{T}$ for each $\mathsf{a} \in \overline{\mathcal{D}^{\sim}_{\alpha}}$, iff $\upsilon^{\sim}(\mathsf{P}@^{\sim}\mathsf{A}) \equiv \mathtt{T}$ for each $\mathsf{A} \in \mathcal{D}^{\sim}_{\alpha}$. This last equivalence is true since for every $\mathsf{a} \in \overline{\mathcal{D}^{\sim}_{\alpha}}$, we can use $[\![\mathsf{a}]\!]_{\sim} \in \mathcal{D}^{\sim}_{\alpha}$ with $\mathsf{a} \sim [\![\mathsf{a}]\!]^{*}_{\sim}$ to determine $\upsilon(\mathsf{P}^{*}@\mathsf{a}) \equiv \mathtt{T}$, and for every $\mathsf{A} \in \mathcal{D}^{\sim}_{\alpha}$, we can use $\mathsf{A}^{*} \in \overline{\mathcal{D}^{\sim}_{\alpha}}$ to determine $\upsilon^{\sim}(\mathsf{P}@^{\sim}\mathsf{A}) \equiv \mathtt{T}$.

So, we have the desired Σ-model $\mathcal{M}$. Now, we check the other properties.

1. Suppose $\mathcal{P}$ satisfies $\partial^{\mathfrak{q}}$. We need to show $\mathcal{M}$ satisfies $\mathfrak{q}$. Let α be a type and $\mathrm{q}^{\alpha} \in \overline{\mathcal{D}^{\sim}_{\alpha\to\alpha\to o}}$ be the element guaranteed to exist by $\partial^{\mathfrak{q}}$. We will show $[\![\mathrm{q}]\!]_{\sim} \in \mathcal{D}^{\sim}_{\alpha\to\alpha\to o}$ is the required witness for $\mathfrak{q}$. Let $\mathsf{A}, \mathsf{B} \in \mathcal{D}^{\sim}_{\alpha}$ be given. We have $\upsilon^{\sim}([\![\mathrm{q}]\!]_{\sim}@^{\sim}\mathsf{A}@^{\sim}\mathsf{B}) \equiv \mathtt{T}$, iff $\upsilon(\mathrm{q}@\mathsf{A}^{*}@\mathsf{B}^{*}) \equiv \mathtt{T}$, iff (by $\partial^{\mathfrak{q}}$) $\mathsf{A}^{*} \sim \mathsf{B}^{*}$, iff $\mathsf{A} \equiv \mathsf{B}$. So, if $\mathcal{P}$ satisfies $\partial^{\mathfrak{q}}$, then $\mathcal{M}$ satisfies $\mathfrak{q}$.
2. Suppose $\mathcal{P}$ satisfies ∂^{η}. To check that the model $\mathcal{M}$ satisfies property η, let $\mathbf{A} \in \mathit{wff}_{\alpha}(\Sigma)$ and an assignment φ for $\mathcal{M}$ be given. By ∂^{η}, we have $\mathcal{E}_{\varphi^{*}}(\mathbf{A}) \sim \mathcal{E}_{\varphi^{*}}(\mathbf{A}{\downarrow}_{\beta\eta})$ and so $\mathcal{E}^{\sim}_{\varphi}(\mathbf{A}) \equiv \mathcal{E}^{\sim}_{\varphi}(\mathbf{A}{\downarrow}_{\beta\eta})$.
3. Suppose $\mathcal{P}$ satisfies ∂^{ξ}. To check property ξ, suppose $\mathbf{M}, \mathbf{N} \in \mathit{wff}_{\beta}$, φ is an assignment into $\mathcal{D}^{\sim}$, X_{α} is a variable and $\mathcal{E}^{\sim}_{\varphi,[\mathsf{A}/X]}(\mathbf{M}) \equiv \mathcal{E}^{\sim}_{\varphi,[\mathsf{A}/X]}(\mathbf{N})$ for every $\mathsf{A} \in \mathcal{D}^{\sim}_{\alpha}$. This implies $\mathcal{E}_{\varphi^{*},[\mathsf{a}/X]}(\mathbf{M}) \sim \mathcal{E}^{\sim}_{\varphi^{*},[\mathsf{a}/X]}(\mathbf{N})$ for every $\mathsf{a} \in \overline{\mathcal{D}^{\sim}_{\alpha}}$. Since $\varphi^{*} \sim \varphi^{*}$ we conclude $\mathcal{E}_{\varphi^{*}}(\lambda X.\mathbf{M}) \sim \mathcal{E}^{\sim}_{\varphi^{*}}(\lambda X.\mathbf{N})$ from ∂^{ξ}. That is, $\mathcal{E}^{\sim}_{\varphi}(\lambda X_{\alpha}.\mathbf{M}) \equiv \mathcal{E}^{\sim}_{\varphi}(\lambda X_{\alpha}.\mathbf{N})$.
4. Suppose $\mathcal{P}$ satisfies $\partial^{\mathfrak{f}}$. It easily follows that $\mathcal{M}$ is functional. Let $\mathsf{G}, \mathsf{H} \in \mathcal{D}^{\sim}_{\alpha\to\beta}$ be such that for every $\mathsf{A} \in \mathcal{D}^{\sim}_{\alpha}$ we have $\mathsf{G}@^{\sim}\mathsf{A} \equiv \mathsf{H}@^{\sim}\mathsf{A}$. This implies $\mathsf{G}^{*}@\mathsf{a} \sim \mathsf{H}^{*}@\mathsf{a}$ for every $\mathsf{a} \in \overline{\mathcal{D}^{\sim}_{\alpha}}$. If we take any $\mathsf{a}, \mathsf{b} \in \mathcal{D}_{\alpha}$ with $\mathsf{a} \sim \mathsf{b}$, then we have $\mathsf{G}^{*}@\mathsf{a} \sim \mathsf{H}^{*}@\mathsf{a} \sim \mathsf{H}^{*}@\mathsf{b}$ by $\partial^{\mathfrak{f}}$ since $\mathsf{H}^{*} \in \overline{\mathcal{D}^{\sim}_{\alpha\to\beta}}$. By $\partial^{\mathfrak{f}}$ again, we have $\mathsf{G}^{*} \sim \mathsf{H}^{*}$. But this implies $\mathsf{G} \equiv \mathsf{H}$, as desired.
5. Finally, suppose $\mathcal{P}$ satisfies $\partial^{\mathfrak{b}}$. Then we immediately have by the definition of $\mathcal{D}^{\sim}_{o}$ that this domain has only two elements. So, $\mathcal{M}$ satisfies property $\mathfrak{b}$.

□

2.4.4 Model Existence Theorem

Suppose Γ_Σ is an acceptable abstract consistency class in $\mathfrak{Acc}_*$ and $\Phi \in \Gamma_\Sigma$. Our goal in this section is to find a model in $\mathfrak{M}_*$ with $\mathcal{M} \models \Phi$. This will be used to prove the saturated extension theorem. The saturated abstract consistency class $\Gamma_\Sigma^{\mathfrak{M}_*}$ will be an extension of any acceptable, sufficiently pure abstract consistency class in $\mathfrak{Acc}_*$ because of the following model existence theorem.

Theorem 2.4.11 (Model Existence for $\mathfrak{Acc}_*$) *Let Γ_Σ be an acceptable $\mathfrak{Acc}_*$, and let $\Phi \in \Gamma_\Sigma$ be a sufficiently Σ-pure set of Σ-sentences. There exists a model $\mathcal{M} \in \mathfrak{M}_*$, such that $\mathcal{M} \models \Phi$.*

The proof of this theorem will be given in Section 2.4.11. Before we can give we need to develop single model existence results the different kinds of acceptable abstract consistency classes.

2.4.5 Possible Values Structures

In this subsection, we develop in the abstract the structures which will be used to construct models.

Definition 2.4.12 (Possible Values Structure) A *[η-]possible values structure* $\mathcal{F} \equiv (\mathcal{D}, @)$ is an applicative structure $(\mathcal{D}, @)$ where

1. For each type α, $\mathsf{a} \in \mathcal{D}_\alpha$ implies $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle$ where $\mathbf{A} \in cwff_\alpha(\Sigma)$ is β-normal [$\beta\eta$-normal].
2. At each base type α, for every $\mathbf{A} \in cwff_\alpha(\Sigma)$ in β-normal form [$\beta\eta$-normal form], there exists an a with $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$.
3. For each function type $\alpha \to \beta$, $\langle \mathbf{G}, g\rangle \in \mathcal{D}_\alpha \to \beta$ iff $\mathbf{G} \in cwff_{\alpha\to\beta}(\Sigma)$ is β-normal [$\beta\eta$-normal], $g : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ and for every $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$ the first component of $g(\langle \mathbf{A}, a\rangle)$ is the β-normal form [$\beta\eta$-normal form] of $\mathbf{GA}$.
4. For each $\langle \mathbf{G}, g\rangle \in \mathcal{D}_{\alpha\to\beta}$ and $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$, $\langle \mathbf{G}, g\rangle @\langle \mathbf{A}, a\rangle \equiv g(\langle \mathbf{A}, a\rangle)$ where $\mathbf{B}$ is the β-normal form [$\beta\eta$-normal form] of $\mathbf{GA}$.

We use the notation $\mathbf{A}{\downarrow_*}$ to mean the β-normal form in the β case and $\beta\eta$-normal form in the $\beta\eta$ case. We also use the term "normal" ambiguously to consider both cases.

Definition 2.4.13 (Possible Value) Let $\mathcal{A} \equiv (\mathcal{D}, @, \mathcal{E})$ be a [η-]possible values structure. We will call a a *possible value* for $\mathbf{A} \in cwff_\alpha(\Sigma)$ if $\langle \mathbf{A}{\downarrow_*}, a\rangle \in \mathcal{D}_\alpha^{\mathbf{A}} \subseteq \mathcal{D}_\alpha$.

Lemma 2.4.14 (Possible Values Exist) *Let $\mathcal{F}$ be a [η-]possible values structure for $\mathcal{H}$. For each closed term $\mathbf{A} \in cwff_\alpha(\Sigma)$, there is a possible value p for $\mathbf{A}$ in $\mathcal{F}$.*

Proof: We prove this by induction on the type α as in [And71].

$\alpha \in \{o, \iota\}$: By the definition of a possible values structure, we know there is an a with $\langle \mathbf{A}{\downarrow_*}, a\rangle \in \mathcal{D}_\alpha$. This a is the desired possible value.

$\beta \to \gamma$: By induction, there are possible values $p^{\mathbf{AB}}$ for $\mathbf{AB}$ for each $\langle \mathbf{B}, b\rangle \in \mathcal{D}_\beta$. Using the axiom of choice (at the metalevel), there is a function $p : \mathcal{D}_\beta \to \mathcal{D}_\gamma$ such that $p(\langle \mathbf{B}, b\rangle) \equiv \langle (\mathbf{AB}){\downarrow_*}, p^{\mathbf{AB}}\rangle$. This guarantees p is a possible value for $\mathbf{A}$.

$\square$

Since we are interested in interpreting a term $\mathbf{A} \in \mathit{cwff}_\alpha(\Sigma)$ to be a pair of the form $\langle \mathbf{A}{\Downarrow_*}, a\rangle$ it is helpful to adopt a notation to reflect this. So, let

$$\mathcal{D}_\alpha^{\mathbf{A}} := \{\langle \mathbf{B}, a\rangle \in \mathcal{D}_\alpha | \mathbf{B} \equiv \mathbf{A}{\Downarrow_*}\}$$

Remark 2.4.15 Note that the syntactic condition on $\mathcal{D}_{\alpha\to\beta}$ guarantees that if $\langle \mathbf{G}, g\rangle \in \mathcal{D}_{\alpha\to\beta}$, then $g : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ restricts to a mapping $g : \mathcal{D}_\alpha^{\mathbf{A}} \to \mathcal{D}_\beta^{\mathbf{GA}}$ for any $\mathbf{A} \in \mathit{cwff}_\alpha(\Sigma)$.

Definition 2.4.16 (Possible Values Evaluation) A *[η-]possible values evaluation* is a Σ-evaluation $\mathcal{A} \equiv (\mathcal{D}, @, \mathcal{E})$ where $(\mathcal{D}, @)$ is a [η-]possible values structure and $\mathcal{E}(c) \in \mathcal{D}_\alpha^c$ for each constant $c \in \Sigma_\alpha$.

Let φ be a variable assignment. Note that we can extract a substitution θ^φ from φ by taking $\theta^\varphi(x) := \mathbf{A}$ whenever $\varphi(x) \equiv \langle \mathbf{A}, a\rangle$. Also, for each variable x_α, $\theta^\varphi(x)$ is a closed normal term of type α.

We can always extend an interpretation of constants in a possible values structure to obtain a possible values evaluation.

Theorem 2.4.17 *Let $\mathcal{A} \equiv (\mathcal{D}, @)$ be a [η-]possible values structure and $\mathcal{I} : \Sigma \longrightarrow \mathcal{D}$ be a typed function such that $\mathcal{I}(c) \in \mathcal{D}_\alpha^c$ for every $c \in \Sigma_\alpha$. There is an evaluation function $\mathcal{E}$ so that $(\mathcal{D}, @, \mathcal{E})$ is a [η-]possible values evaluation with $\mathcal{E}|_\Sigma \equiv \mathcal{I}$. Also, $\mathcal{E}_\varphi(\mathbf{A}) \in \mathcal{D}_\alpha^{\theta^\varphi(\mathbf{A})}$ for each $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, assignment φ and substitution θ^φ where $\theta^\varphi(x)$ is the first component of $\varphi(x)$ for every variable x. Furthermore, if $\mathcal{A}$ is an η-possible values structure, then $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{A}{\downarrow_{\beta\eta}})$ for every $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$ and assignment φ.*

Proof: We extend $\mathcal{I}$ to an evaluation function $\mathcal{E}$ by induction on terms. At each stage, we ensure that the first component of $\mathcal{E}_\varphi(\mathbf{A}_\alpha)$ is $\theta^\varphi(\mathbf{A}){\Downarrow_*}$, i.e., $\mathcal{E}_\varphi(\mathbf{A}) \in \mathcal{D}_\alpha^{\theta^\varphi(\mathbf{A})}$.

- For each $c_\alpha \in \Sigma$, we must let $\mathcal{E}_\varphi(c) := \mathcal{I}(c) \in \mathcal{D}_\alpha^c$.
- For variables x_α, let $\mathcal{E}_\varphi(x) := \varphi(x) \in \mathcal{D}_\alpha^{\theta^\varphi(x)}$. This ensures that $\mathcal{E}$ satisfies the first condition to be an evaluation function.
- For application, we have $\mathcal{E}_\varphi(\mathbf{G}) \equiv \langle \theta^\varphi(\mathbf{G}){\Downarrow_*}, g\rangle$ where $g : \mathcal{D}_\alpha^{\theta^\varphi(\mathbf{A})} \to \mathcal{D}_\beta^{\theta^\varphi(\mathbf{G})\theta^\varphi(\mathbf{A})}$ and $\mathcal{E}_\varphi(\mathbf{A}) \equiv \langle \theta^\varphi(\mathbf{A}){\Downarrow_*}, a\rangle$. Let $\mathcal{E}_\varphi(\mathbf{GA}) := \mathcal{E}_\varphi(\mathbf{G}) @ \mathcal{E}_\varphi(\mathbf{A})$. This definition ensures the second condition for $\mathcal{E}$ to be an evaluation function. Also, $\mathcal{E}_\varphi(\mathbf{GA}) \equiv g(\langle \mathbf{A}, a\rangle) \in \mathcal{D}_\beta^{\theta^\varphi(\mathbf{GA})}$ guarantees the first component of $\mathcal{E}_\varphi(\mathbf{GA})$ is $(\theta^\varphi(\mathbf{GA})){\Downarrow_*}$
- For abstraction, suppose we have $\mathcal{E}_{\varphi,[\langle \mathbf{A},a\rangle/X]}(\mathbf{B}) \in \mathcal{D}_\beta^{\theta^\varphi([\mathbf{A}/X]\mathbf{B})}$ for each $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$. This defines a function g from $\mathcal{D}_\alpha$ to $\mathcal{D}_\beta$ which properly restricts as $g : \mathcal{D}_\alpha^{\theta^\varphi(\mathbf{A})} \to \mathcal{D}_\beta^{\theta^\varphi([\mathbf{A}/X]\mathbf{B})}$. Let $\mathcal{E}_\varphi(\lambda X_\alpha.\mathbf{B}) := \langle (\lambda X_\alpha.\mathbf{B}){\Downarrow_*}, g\rangle \in \mathcal{D}_{\alpha\to\beta}^{\theta^\varphi(\lambda X.\mathbf{B})}$. Note that we have chosen the first component to be $(\lambda X_\alpha.\mathbf{B}){\Downarrow_*}$, thus maintaining this invariant.

To complete the verification that $\mathcal{E}$ is an evaluation function, we need to check two more conditions.

- Suppose φ and ψ coincide on $\mathit{free}(\mathbf{A})$. An easy induction using the definition of $\mathcal{E}$ shows $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\psi(\mathbf{A})$.

- To show $\mathcal{E}$ respects β-reduction, we show $\mathcal{E}$ respects a single reduction, then use induction on the number of reductions.

  First, we show $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$ by induction on $\mathbf{A}$. If $\mathbf{A}$ is X, then $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(X) \equiv \mathcal{E}_\varphi(\mathbf{B}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]X)$. If $\mathbf{A}$ is a constant or any variable other than X, then $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$. If $\mathbf{A}$ is an application $\mathbf{FC}$, then the induction hypothesis implies

  $$\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{FC}) \equiv \mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{F})@\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{C}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{F})@\mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{C}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{FC})$$

  If $\mathbf{A}$ is an abstraction $\lambda Y_\beta.\mathbf{C}_\gamma$, then we check that the first and second components are equal. The first component of $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A})$ is $(\theta^\varphi, [\theta^\varphi(\mathbf{B})\!\downarrow_* /X])(\mathbf{A})\!\downarrow_*$. This simplifies to $\theta^\varphi([\mathbf{B}/X]\mathbf{A})\!\downarrow_*$, which is the first component of $\mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$. So, we know the first components are equal. The second component of $\mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A})$ is the function $g : \mathcal{D}_\beta \longrightarrow \mathcal{D}_\gamma$ such that $g(\mathsf{b}) \equiv \mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X],[\mathsf{b}/Y]}(\mathbf{C})$ for every $\mathsf{b} \in \mathcal{D}_\beta$. The second component of $\mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$ is the function $h : \mathcal{D}_\beta \longrightarrow \mathcal{D}_\gamma$ such that $h(\mathsf{b}) \equiv \mathcal{E}_{\varphi,[\mathsf{b}/Y]}([\mathbf{B}/X]\mathbf{C})$ for every $\mathsf{b} \in \mathcal{D}_\beta$. The induction hypothesis implies $g(\mathsf{b}) \equiv h(\mathsf{b})$ for every $\mathsf{b} \in \mathcal{D}_\beta$. That is, $g \equiv h$. Hence, the second components are also equal and we are done.

  Now, using the definition of $\mathcal{E}$ on applications and abstractions, we have

  $$\mathcal{E}_\varphi((\lambda X.\mathbf{A})\mathbf{B}) \equiv \mathcal{E}_\varphi(\lambda X.\mathbf{A})@\mathcal{E}_\varphi(\mathbf{B}) \equiv \mathcal{E}_{\varphi,[\mathcal{E}_\varphi(\mathbf{B})/X]}(\mathbf{A}) \equiv \mathcal{E}_\varphi([\mathbf{B}/X]\mathbf{A})$$

  Next, if $\mathbf{C}$ β-reduces to $\mathbf{D}$ in a single step, then induction on the position of the redex in $\mathbf{C}$ shows $\mathcal{E}_\varphi(\mathbf{C}) \equiv \mathcal{E}_\varphi(\mathbf{D})$. Finally, induction on the number of β-reduction steps shows $\mathcal{E}_\varphi(\mathbf{C}) \equiv \mathcal{E}_\varphi(\mathbf{C}\!\downarrow_\beta)$.

In case $\mathcal{A}$ is an η-possible values structure, we can show $\mathcal{E}_\varphi(\lambda X_\beta.\mathbf{A}_{\alpha\to\beta}X) \equiv \mathcal{E}_\varphi(\mathbf{A})$ if $X \notin free(\mathbf{A})$. The first components are equal using the invariant and the fact that $\theta^\varphi(\lambda X.\mathbf{A}X)\!\downarrow_{\beta\eta} \equiv \theta^\varphi(\mathbf{A})\!\downarrow_{\beta\eta}$. Let $g : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ be the second component of $\mathcal{E}_\varphi(\lambda X_\beta.\mathbf{A}X)$ and $h : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ be the second component of $\mathcal{E}_\varphi(\mathbf{A})$. By definition of $\mathcal{E}_\varphi$, we know

$$g(\mathsf{a}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{A}X) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{A})@\mathcal{E}_{\varphi,[\mathsf{a}/X]}(X) \equiv \mathcal{E}_\varphi(\mathbf{A})@\mathsf{a}$$

By definition of @, we know $\mathcal{E}_\varphi(\mathbf{A})@\mathsf{a} \equiv h(\mathsf{a})$. So, $g \equiv h$.

Now, just as in the β-reduction case, we can show that $\mathcal{E}_\varphi(\mathbf{C}) \equiv \mathcal{E}_\varphi(\mathbf{D})$ whenever $\mathbf{C}$ η-reduces to $\mathbf{D}$ in one step by induction on the η-redex in $\mathbf{C}$. Then we have $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\varphi(\mathbf{A}\!\downarrow_{\beta\eta})$ by induction on the number of $\beta\eta$-reductions. □

2.4.6 Model Existence for $\mathfrak{Acc}_\beta$ and $\mathfrak{Acc}_{\beta\eta}$

Definition 2.4.18 ($\mathcal{H}$-Possible Booleans) Let $\mathcal{H} \in \mathfrak{Hint}_*$ and $\mathbf{A} \in cwff_o(\Sigma)$. We define the set $\mathcal{B}^{\mathbf{A}}_{\mathcal{H}}$ of *$\mathcal{H}$-possible booleans* for $\mathbf{A}$ by

- $\mathcal{B}^{\mathbf{A}}_{\mathcal{H}} := \{\mathtt{T}, \mathtt{F}\}$ if $\mathbf{A} \notin \mathcal{H}$ and $\neg\mathbf{A} \notin \mathcal{H}$,
- $\mathcal{B}^{\mathbf{A}}_{\mathcal{H}} := \{\mathtt{T}\}$ if $\mathbf{A} \in \mathcal{H}$, and
- $\mathcal{B}^{\mathbf{A}}_{\mathcal{H}} := \{\mathtt{F}\}$ if $\neg\mathbf{A} \in \mathcal{H}$.

Note that by ∇_c, we cannot have $\mathbf{A}, \neg\mathbf{A} \in \mathcal{H}$. So, $\mathcal{B}^{\mathbf{A}}_{\mathcal{H}}$ is defined for every $\mathbf{A} \in cwff_o(\Sigma)$. Note also that each $\mathcal{B}^{\mathbf{A}}_{\mathcal{H}}$ is nonempty.

Definition 2.4.19 (Andrews Structure) Let $\mathcal{H} \in \mathfrak{Hint}_*$ and let $*$ be some fixed arbitrary value. We define a possible values structure we will call the *[η-]Andrews Structure* $\mathcal{F}^{\mathcal{H}} := (\mathcal{D}, @)$ $[\mathcal{F}^{\mathcal{H}}_{\eta} := (\mathcal{D}, @)]$ by

- Let $\mathcal{D}_o$ be the set of pairs $\langle \mathbf{A}_o, p\rangle$ where $\mathbf{A} \in cwff_o(\Sigma)$ is β-normal [$\beta\eta$-normal] and $p \in \mathcal{B}^{\mathbf{A}}_{\mathcal{H}}$.
- Let $\mathcal{D}_\iota$ be the set of pairs $\langle \mathbf{A}_\iota, *\rangle$ where $\mathbf{A} \in cwff_\iota(\Sigma)$ is β-normal [$\beta\eta$-normal].
- Let $\mathcal{D}_{\alpha\to\beta}$ be the set of pairs $\langle \mathbf{G}, g\rangle$ where $\mathbf{G} \in cwff_{\alpha\to\beta}(\Sigma)$ is β-normal [$\beta\eta$-normal] and $g : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ such that for every $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$, $g(\langle \mathbf{A}, a\rangle) \equiv \langle \mathbf{B}, b\rangle$ implies $\mathbf{B}$ is the β-normal form [$\beta\eta$-normal form] of $\mathbf{GA}$.

We define @ by

$$\langle \mathbf{G}, g\rangle @ \langle \mathbf{A}, a\rangle \equiv g(\langle \mathbf{A}, a\rangle)$$

where $\mathbf{B}$ is the β-normal form [$\beta\eta$-normal form] of $\mathbf{GA}$ for each $\langle \mathbf{G}, g\rangle \in \mathcal{D}_{\alpha\to\beta}$ and $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$.

It is trivial to check most of the conditions that the Andrews Structure is a possible values structure, and the η-Andrews Structure is an η-possible values structure.

We can argue the $\mathfrak{Acc}_\beta$ and $\mathfrak{Acc}_{\beta\eta}$ simultaneously, again using the notation $\mathbf{A}{\downarrow_*}$ to mean the β-normal form of $\mathbf{A}$ in the $\mathfrak{Acc}_\beta$ case and the $\beta\eta$-normal form of $\mathbf{A}$ in the $\mathfrak{Acc}_{\beta\eta}$ case. When we use the term "normal" in this section, we mean β-normal in the $\mathfrak{Acc}_\beta$ case and the normal in the $\mathfrak{Acc}_{\beta\eta}$ case.

Theorem 2.4.20 (Model Existence ($\mathfrak{Acc}_\beta$, $\mathfrak{Acc}_{\beta\eta}$)) *Let $\mathcal{H}$ be a Hintikka set in $\mathfrak{Hint}_\beta$ [$\mathfrak{Hint}_{\beta\eta}$]. There is a model $\mathcal{M}$ in $\mathfrak{M}_\beta$ [$\mathfrak{M}_{\beta\eta}$] such that $\mathcal{M} \models \mathcal{H}$.*

Proof: Here we directly follow Andrews v-complex construction [And71]. Let $\mathcal{F} \equiv (\mathcal{D}, @)$ be the [η-]Andrews Structure. By Lemma 2.4.14, every closed term has a possible value. For each parameter $c_\alpha \in \Sigma$, choose p^c to be some possible value for c, so $\langle c, p^c\rangle \in \mathcal{D}_\alpha$. For the logical constants $\neg$, $\vee$ and Π^α, we must take particular values.

We define values for the logical constants as follows.

$\neg$ Let $p^\neg : \mathcal{D}_o \to \mathcal{D}_o$ be defined by $p^\neg(\langle \mathbf{A}, a\rangle) := \langle \neg\mathbf{A}, b\rangle$ where b is $\mathtt{T}$ if a is $\mathtt{F}$ and b is $\mathtt{F}$ if a is $\mathtt{T}$. The $\overline{\nabla}_\neg$ and ∇_c properties of $\mathcal{H}$ guarantees this is well-defined. So, $p^\neg$ is a possible value for $\neg$.

$\vee$ For each $\langle \mathbf{A}, \mathtt{F}\rangle \in \mathcal{D}_o$, let $p^\vee_{\langle \mathbf{A},\mathtt{F}\rangle} : \mathcal{D}_o \to \mathcal{D}_o$ be the function defined by $p^\vee_{\langle \mathbf{A},\mathtt{F}\rangle}(\langle \mathbf{B}, b\rangle) := \langle \mathbf{A}\vee\mathbf{B}, b\rangle$. For each $\langle \mathbf{A}, \mathtt{T}\rangle \in \mathcal{D}_o$, let $p^\vee_{\langle \mathbf{A},\mathtt{T}\rangle} : \mathcal{D}_o \to \mathcal{D}_o$ be the function defined by $p^\vee_{\langle \mathbf{A},\mathtt{T}\rangle}(\langle \mathbf{B}, b\rangle) := \langle \mathbf{A} \vee \mathbf{B}, \mathtt{T}\rangle$. The properties $\overline{\nabla}_\vee$, $\overline{\nabla}_\wedge$ and ∇_c of $\mathcal{H}$ guarantees these are well-defined and $\langle \vee\mathbf{A}, p^\vee_{\langle \mathbf{A},a\rangle}\rangle \in \mathcal{D}_{o\to o}$. Now, let $p^\vee : \mathcal{D}_o \to \mathcal{D}_{o\to o}$ be the function defined by $p^\vee(\langle \mathbf{A}, a\rangle) := \langle \vee\mathbf{A}, p^\vee_{\langle \mathbf{A},a\rangle}\rangle$. Clearly, $p^\vee$ is a possible value for $\vee$.

Π^α Let $p^{\Pi^\alpha} : \mathcal{D}_{\alpha\to o} \to \mathcal{D}_o$ be the function defined by $p^{\Pi^\alpha}(\mathbf{F}, f) := \langle \Pi^\alpha\mathbf{F}, p\rangle$ where $p \equiv \mathtt{T}$ if for every $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$, the second component of $f(\langle \mathbf{A}, a\rangle)$ is $\mathtt{T}$, and $p \equiv \mathtt{F}$ otherwise. This is well-defined by $\overline{\nabla}_\forall$, $\overline{\nabla}_\exists$ and ∇_c, and p^{Π^α} is a possible value for Π^α.

Define $\mathcal{I}(c) := \langle c, p^c\rangle$ for each $c \in \Sigma$. Let $\mathcal{E}$ be the evaluation function extending $\mathcal{I}$ guaranteed to exist by Theorem 2.4.17 so that $(\mathcal{D}, @, \mathcal{E})$ is a [η-]possible values evaluation.

To make this a Σ-model, we must define a valuation $v : \mathcal{D}_o \longrightarrow \{\mathtt{T}, \mathtt{F}\}$. We take the obvious choice $v(\langle \mathbf{A}, p\rangle) := p$. Let $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, v)$. To check $\mathcal{M}$ is a Σ-model, we must check the requirements for v.

$\neg$: $v(\mathcal{E}(\neg)@\mathsf{a}) \equiv \mathtt{T}$, iff $v(\mathsf{a}) \equiv \mathtt{F}$ by the definition of $p^{\neg}$.

$\vee$: $v(\mathcal{E}(\vee)@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$, iff $v(\mathsf{a}) \equiv \mathtt{T}$ or $v(\mathsf{b}) \equiv \mathtt{T}$ by the definition of $p^{\vee}$.

Π: $v(\mathcal{E}(\Pi^{\alpha})@\mathsf{f}) \equiv \mathtt{T}$, iff $v(\mathsf{f}@\mathsf{a}) \equiv \mathtt{T}$ for each $\mathsf{a} \in \mathcal{D}_{\alpha}$ by the definition of $p^{\Pi^{\alpha}}$.

To see that $\mathcal{M} \models \mathcal{H}$, note that for any $\mathbf{A} \in \mathcal{H}$, we have $\mathcal{E}(\mathbf{A}) \equiv \langle \mathbf{A}{\downarrow_*}, p\rangle \in \mathcal{D}_o$. Since $\mathbf{A} \in \mathcal{H}$, the definition of $\mathcal{D}_o$ forces p to be $\mathtt{T}$. So, $\mathcal{M} \models \mathbf{A}$.

In the $\mathfrak{Hint}_{\beta\eta}$ case, Theorem 2.4.17 guarantees $\mathcal{M}$ satisfies property η.

In general, we can use Theorem 1.3.62 to obtain a model of $\mathcal{H}$ satisfying property $\mathfrak{q}$. However, we get more information by using the Hintikka Gap (Theorem ??). If $\mathcal{H}$ is complete, we already constructed a term model modulo Leibniz in Theorem ??. If $\mathcal{H}$ is equation-free, we can show the possible values model $\mathcal{M}$ *already* satisfies property $\mathfrak{q}$, so there is no need to take a quotient via Leibniz. To see this, for each $\langle \mathbf{A}, a\rangle \in \mathcal{D}_{\alpha}$, let $s_{\langle \mathbf{A},a\rangle} : \mathcal{D}_{\alpha} \to \mathcal{D}_o$ be defined by $s_{\langle \mathbf{A},a\rangle}(\langle \mathbf{A}, a\rangle) := \langle (\mathbf{A} \doteq \mathbf{A}){\downarrow_*}, \mathtt{T}\rangle$ and $s_{\langle \mathbf{A},a\rangle}(\langle \mathbf{B}, b\rangle) := \langle (\mathbf{A} \doteq \mathbf{B}){\downarrow_*}, \mathtt{F}\rangle$ for $\langle \mathbf{B}, b\rangle \not\equiv \langle \mathbf{A}, a\rangle$. This is well-defined since we never have $\neg(\mathbf{A} \doteq \mathbf{A}){\downarrow_*} \in \mathcal{H}$, and $(\mathbf{A} \doteq \mathbf{B}){\downarrow_*} \notin \mathcal{H}$ since $\mathcal{H}$ is equation-free. Then, define $\mathrm{q}^{\alpha} := \langle \doteq^{\alpha}, l\rangle$ where $l(\langle \mathbf{A}, a\rangle) := \langle (\lambda X\boldsymbol{.}\mathbf{A} \doteq x){\downarrow_*}, s_{\langle \mathbf{A},a\rangle}\rangle$. This witnesses $\mathcal{M}$ satisfies property $\mathfrak{q}$. $\square$

2.4.7 $\mathcal{H}$-Functional Compatibility

Remark 2.4.21 We use $\mathbf{A} \doteq \mathbf{B}$ to denote the β-reduct $\forall Q\boldsymbol{.}\neg(Q\mathbf{A}) \vee (Q\mathbf{B})$.

Given a Hintikka set $\mathcal{H}$, we can define a functional compatibility relation. We will use this compatibility relation to prove the existence of models in $\mathfrak{M}_{\beta\mathfrak{f}}$ and $\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$.

Definition 2.4.22 ($\mathcal{H}$-Functional Compatibility) Let $\mathcal{H}$ be a Hintikka set. We define when two closed terms $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_{\alpha}(\Sigma)$ are *$\mathcal{H}$-functionally compatible* (written $\mathbf{A} \|^{\mathfrak{f}} \mathbf{B}$) by induction on the type α

o $\mathbf{A}_o \|^{\mathfrak{f}} \mathbf{B}_o$ if $\neg(\mathbf{A}{\downarrow_\beta} \doteq^{o} \mathbf{B}{\downarrow_\beta}) \notin \mathcal{H}$, $\{\mathbf{A}{\downarrow_\beta}, \neg\mathbf{B}{\downarrow_\beta}\} \not\subseteq \mathcal{H}$ and $\{\neg\mathbf{A}{\downarrow_\beta}, \mathbf{B}{\downarrow_\beta}\} \not\subseteq \mathcal{H}$.

ι $\mathbf{A}_{\iota} \|^{\mathfrak{f}} \mathbf{B}_{\iota}$ if $\neg(\mathbf{A}{\downarrow_\beta} \doteq^{\iota} \mathbf{B}{\downarrow_\beta}) \notin \mathcal{H}$.

$\beta \to \gamma$: $\mathbf{A}_{\beta\to\gamma} \|^{\mathfrak{f}} \mathbf{B}_{\beta\to\gamma}$ if $\mathbf{AC} \|^{\mathfrak{f}} \mathbf{BD}$ whenever $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_{\beta}(\Sigma)$ and $\mathbf{C} \|^{\mathfrak{f}} \mathbf{D}$.

We will say a set $S \subseteq \mathit{cwff}_{\alpha}(\Sigma)$ is $\mathcal{H}$-functionally compatible if $\mathbf{A} \|^{\mathfrak{f}} \mathbf{B}$ for every pair $\mathbf{A}, \mathbf{B} \in S$.

Intuitively, if two terms $\mathcal{H}$-functionally compatible, then they *might* be equal in a functionally extensional model of $\mathcal{H}$. Notice that $\mathcal{H}$-functional compatibility is not an equivalence relation, as it need not be transitive.

We also use a notion of $\mathcal{H}$-compatibility for $\mathfrak{Hint}_{\beta\xi}$ and $\mathfrak{Hint}_{\beta\xi\mathfrak{b}}$.

Definition 2.4.23 ($\mathcal{H}$-ξ-Compatibility) Let $\mathcal{H}$ be a Hintikka set. We define when two closed terms $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_{\alpha}(\Sigma)$ are *$\mathcal{H}$-ξ-compatible* (written $\mathbf{A} \|^{\xi} \mathbf{B}$) by induction on the type α

o $\mathbf{A}_o \|^{\xi} \mathbf{B}_o$ if $\neg(\mathbf{A}{\downarrow_\beta} \doteq^{o} \mathbf{B}{\downarrow_\beta}) \notin \mathcal{H}$, $\{\mathbf{A}{\downarrow_\beta}, \neg\mathbf{B}{\downarrow_\beta}\} \not\subseteq \mathcal{H}$ and $\{\neg\mathbf{A}{\downarrow_\beta}, \mathbf{B}{\downarrow_\beta}\} \not\subseteq \mathcal{H}$.

ι $\mathbf{A}_{\iota} \|^{\xi} \mathbf{B}_{\iota}$ if $\neg(\mathbf{A}{\downarrow_\beta} \doteq^{\iota} \mathbf{B}{\downarrow_\beta}) \notin \mathcal{H}$.

$\beta \to \gamma$: $\mathbf{A}_{\beta\to\gamma} \|_{\xi}\, \mathbf{B}_{\beta\to\gamma}$ if $\neg(\mathbf{A}{\downarrow_{\beta}} \doteq^{\beta\to\gamma} \mathbf{B}{\downarrow_{\beta}}) \notin \mathcal{H}$ and $\mathbf{AC} \|_{\xi}\, \mathbf{BD}$ whenever $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_{\beta}(\Sigma)$ and $\mathbf{C} \|_{\xi}\, \mathbf{D}$

We will say a set $S \subseteq \mathit{cwff}_{\alpha}(\Sigma)$ is $\mathcal{H}$-ξ-compatible if $\mathbf{A} \|_{\xi}\, \mathbf{B}$ for every pair $\mathbf{A}, \mathbf{B} \in S$.

We will use the notation $\|_{\ast}$ to ambiguously denote $\|_{\mathfrak{f}}$ or $\|_{\xi}$.

Lemma 2.4.24 *Let $\mathcal{H}$ be a Hintikka set. For any $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_{\alpha}(\Sigma)$, $\mathbf{A} \|_{\ast}\, \mathbf{B}$ iff $\mathbf{A}{\downarrow_{\beta}} \|_{\ast}\, \mathbf{B}{\downarrow_{\beta}}$.*

Proof: This follows by an easy induction on the type α. At base types, this is trivial since $\|_{\mathfrak{f}}$ and $\|_{\xi}$ are defined in terms of normal forms.

At type $\beta \to \gamma$, let $\mathbf{C}, \mathbf{D} \in \mathit{cwff}_{\beta}(\Sigma)$ with $\mathbf{C} \|_{\ast}\, \mathbf{D}$ be given. By induction, we know $(\mathbf{AC}) \|_{\ast}\, (\mathbf{BD})$ iff $(\mathbf{AC}){\downarrow_{\beta}} \|_{\ast}\, (\mathbf{BD}){\downarrow_{\beta}}$ iff $(\mathbf{A}{\downarrow_{\beta}}\mathbf{C}) \|_{\ast}\, (\mathbf{B}{\downarrow_{\beta}}\mathbf{D})$. For $\|_{\mathfrak{f}}$, we can generalize over $\mathbf{C}$ and $\mathbf{D}$ to conclude $\mathbf{A} \|_{\mathfrak{f}}\, \mathbf{B}$ iff $\mathbf{A}{\downarrow_{\beta}} \|_{\mathfrak{f}}\, \mathbf{B}{\downarrow_{\beta}}$. For $\|_{\xi}$, we also need to check that $\neg(\mathbf{A} \doteq^{\beta\to\gamma} \mathbf{B}) \notin \mathcal{H}$ iff $\neg(\mathbf{A}{\downarrow_{\beta}} \doteq^{\beta\to\gamma} \mathbf{B}{\downarrow_{\beta}}) \notin \mathcal{H}$. But this is immediate from $\vec{\nabla}_{\beta}$. □

We can simplify the definitions of $\|_{\ast}$ at type o in case we have Boolean extensionality.

Lemma 2.4.25 *Let $\mathcal{H}$ be a Hintikka set which satisfies $\vec{\nabla}_{\mathfrak{b}}$. For any $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_{o}(\Sigma)$ with $\{\mathbf{A}{\downarrow_{\beta}}, \neg\mathbf{B}{\downarrow_{\beta}}\} \not\subseteq \mathcal{H}$ and $\{\neg\mathbf{A}{\downarrow_{\beta}}, \mathbf{B}{\downarrow_{\beta}}\} \not\subseteq \mathcal{H}$, we have $\mathbf{A} \|_{\ast}\, \mathbf{B}$.*

Proof: We only need to check $\neg(\mathbf{A} \doteq \mathbf{B}){\downarrow_{\beta}} \notin \mathcal{H}$. But if $\neg(\mathbf{A} \doteq \mathbf{B}){\downarrow_{\beta}} \in \mathcal{H}$, then $\vec{\nabla}_{\mathfrak{b}}$ implies $\{\mathbf{A}{\downarrow_{\beta}}, \neg\mathbf{B}{\downarrow_{\beta}}\} \not\subseteq \mathcal{H}$ or $\{\neg\mathbf{A}{\downarrow_{\beta}}, \mathbf{B}{\downarrow_{\beta}}\} \not\subseteq \mathcal{H}$, contradicting our assumption. □

Lemma 2.4.26 *Let $\mathcal{H}$ be a Hintikka set satisfying $\vec{\nabla}_{m}$. Given two atomic sentences $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_{o}(\Sigma)$, if $\mathbf{A} \|_{\ast}\, \mathbf{B}$ does not hold, then $\neg(\mathbf{A} \doteq^{o} \mathbf{B}){\downarrow_{\beta}} \in \mathcal{H}$.*

Proof: By definition, we have one of three possibilities.

1. $\neg(\mathbf{A} \doteq \mathbf{B}){\downarrow_{\beta}} \in \mathcal{H}$, in which case we are done.

2. $\{\mathbf{A}{\downarrow_{\beta}}, \neg\mathbf{B}{\downarrow_{\beta}}\} \subseteq \mathcal{H}$, in which case we can apply $\vec{\nabla}_{m}$ since $\mathbf{A}$ and $\mathbf{B}$ are atomic.

3. $\{\neg\mathbf{A}{\downarrow_{\beta}}, \mathbf{B}{\downarrow_{\beta}}\} \subseteq \mathcal{H}$, in which case we also apply $\vec{\nabla}_{m}$.

□

Without some form of functional extensionality, there may be constants which are not compatible with themselves (consider q in Example 2.1.2). We can use the assumption that $\mathcal{H}$ is an acceptable Hintikka set in $\mathfrak{Hint}_{\beta\mathfrak{f}}$[24] or $\mathfrak{Hint}_{\beta\mathfrak{f}\mathfrak{b}}$ to show every term is functionally compatible with itself. We will also be able to show this for ξ-compatibility for acceptable Hintikka sets in $\mathfrak{Hint}_{\beta\xi}$ or $\mathfrak{Hint}_{\beta\xi\mathfrak{b}}$. EdNote(24)

Lemma 2.4.27 *Let $\mathcal{H}$ be a Hintikka set satisfying $\vec{\nabla}_{\mathfrak{f}}$, $\vec{\nabla}_{m}$ and $\vec{\nabla}^{w}_{dec}$ for each parameter w.*

1. *If $\mathbf{A} \|_{\mathfrak{f}}\, \mathbf{B}$, then $\neg(\mathbf{A}_{\alpha} \doteq^{\alpha} \mathbf{B}_{\alpha}){\downarrow_{\beta}} \notin \mathcal{H}$.*

[24] EdNote: or xi or xib

2. Given $h_{\alpha_1 \to \cdots \to \alpha_n \to \alpha} \in \Sigma$ (possibly a logical constant), suppose $\mathcal{H}$ satisfies $\vec{\nabla}^h_{dec}$ and for $1 \leq i \leq n$, $\mathbf{C}^i, \mathbf{D}^i \in cwff_{\alpha_i}(\Sigma)$ with $\neg(\mathbf{C}^i \doteq^{\alpha_i} \mathbf{D}^i){\downarrow_\beta} \notin \mathcal{H}$. Then, $(h\overline{\mathbf{C}^n}) \|^{\mathfrak{f}} (h\overline{\mathbf{D}^n})$.

Proof: We can prove this by mutual induction on the type α.

1. $\alpha \in \{o, \iota\}$**:** This is immediate from the definition.

   $\beta \to \gamma$**:** If $\neg(\mathbf{A}_{\beta\to\gamma}{\downarrow_\beta} \doteq \mathbf{B}_{\beta\to\gamma}{\downarrow_\beta}) \in \mathcal{H}$, then by $\vec{\nabla}_{\mathfrak{f}}$ there is a parameter $w \in \Sigma_\beta$ with $\neg(\mathbf{A}w \doteq^\beta \mathbf{B}w) \in \mathcal{H}$. Applying the induction hypothesis for part 2 with w (with no arguments) at the lower type β, we have $w \|^{\mathfrak{f}} w$. So, if $\mathbf{A} \|^{\mathfrak{f}} \mathbf{B}$, then by definition we have $\mathbf{A}w \|^{\mathfrak{f}} \mathbf{B}w$, the induction hypothesis for part 1 at type γ implies $\neg(\mathbf{A}w \doteq^\beta \mathbf{B}w) \notin \mathcal{H}$, a contradiction.

2. $\alpha \in \{\iota, o\}$**:** If $(h\mathbf{C}^1 \cdots \mathbf{C}^n) \|^{\mathfrak{f}} (h\mathbf{D}^1 \cdots \mathbf{D}^n)$ does not hold, then either by definition or by Lemma 2.4.26, we have $\neg((h\mathbf{C}^1{\downarrow_\beta} \cdots \mathbf{C}^n{\downarrow_\beta}) \doteq^\alpha (h\mathbf{D}^1{\downarrow_\beta} \cdots \mathbf{D}^n{\downarrow_\beta})) \in \mathcal{H}$. We can apply $\vec{\nabla}^h_{dec}$ (and $\overline{\nabla}_\beta$) to obtain $\neg(\mathbf{C}^i \doteq^\alpha \mathbf{D}^i){\downarrow_\beta} \in \mathcal{H}$ for some i, contradicting our assumption.

   $\beta \to \gamma$**:** To show $(h\mathbf{C}^1 \cdots \mathbf{C}^n) \|^{\mathfrak{f}} (h\mathbf{D}^1 \cdots \mathbf{D}^n)$, let $\mathbf{A}, \mathbf{B} \in cwff_\beta(\Sigma)$ with $\mathbf{A} \|^{\mathfrak{f}} \mathbf{B}$ be given. Applying the induction hypothesis for part 1 to $\mathbf{A}$ and $\mathbf{B}$ at type β, we know $\neg(\mathbf{A} \doteq^\beta \mathbf{B}){\downarrow_\beta} \notin \mathcal{H}$. So, we can apply the induction hypothesis for part 2 to the two terms $h\mathbf{C}^1 \cdots \mathbf{C}^n\mathbf{A}$ and $h\mathbf{D}^1 \cdots \mathbf{D}^n\mathbf{B}$ at type γ to obtain $(h\mathbf{C}^1 \cdots \mathbf{C}^n\mathbf{A}) \|^{\mathfrak{f}} (h\mathbf{D}^1 \cdots \mathbf{D}^n\mathbf{B})$.

$\square$

In particular, Lemma lem:funccompat:2 shows $w \|^{\mathfrak{f}} w$ for every parameter w. We can simplify the proof of this fact for ξ-compatibility since the definition of ξ-compatibility is stronger at function types.

Lemma 2.4.28 *Let $\mathcal{H}$ be a Hintikka set satisfying $\vec{\nabla}_\xi$, $\vec{\nabla}_m$, $\vec{\nabla}_{\xi dec}$ and $\vec{\nabla}^w_{dec}$ for each parameter w. Given $h_{\alpha_1 \to \cdots \to \alpha_n \to \alpha} \in \Sigma$ (possibly a logical constant), suppose $\mathcal{H}$ satisfies $\vec{\nabla}^h_{dec}$ and for $1 \leq i \leq n$, $\mathbf{C}^i, \mathbf{D}^i \in cwff_{\alpha_i}(\Sigma)$ with $\neg(\mathbf{C}^i \doteq^{\alpha_i} \mathbf{D}^i){\downarrow_\beta} \notin \mathcal{H}$. Then, $(h\overline{\mathbf{C}^n}) \|^{\xi} (h\overline{\mathbf{D}^n})$.*

Proof: We can prove this by induction on the type α.

$\alpha \in \{\iota, o\}$**:** If $(h\mathbf{C}^1 \cdots \mathbf{C}^n) \|^{\xi} (h\mathbf{D}^1 \cdots \mathbf{D}^n)$ does not hold, then either by definition or by Lemma 2.4.26, we have $\neg((h\mathbf{C}^1{\downarrow_\beta} \cdots \mathbf{C}^n{\downarrow_\beta}) \doteq^\alpha (h\mathbf{D}^1{\downarrow_\beta} \cdots \mathbf{D}^n{\downarrow_\beta})) \in \mathcal{H}$. We can apply $\vec{\nabla}^h_{dec}$ (and $\overline{\nabla}_\beta$) to obtain $\neg(\mathbf{C}^i \doteq^\alpha \mathbf{D}^i){\downarrow_\beta} \in \mathcal{H}$ for some i, contradicting our assumption.

$\beta \to \gamma$**:** We need to show $(h\mathbf{C}^1 \cdots \mathbf{C}^n) \|^{\xi} (h\mathbf{D}^1 \cdots \mathbf{D}^n)$. First, let us show $\neg((h\mathbf{C}^1 \cdots \mathbf{C}^n) \doteq (h\mathbf{D}^1 \cdots \mathbf{D}^n)) \notin \mathcal{H}$. If $\neg((h\mathbf{C}^1 \cdots \mathbf{C}^n) \doteq (h\mathbf{D}^1 \cdots \mathbf{D}^n)) \in \mathcal{H}$, then by $\vec{\nabla}_{\xi dec}$ we have $\neg(\mathbf{C}^i \doteq \mathcal{D}^i)$ for some $1 \leq i \leq n$, contradicting our assumption.

Next, let $\mathbf{A}, \mathbf{B} \in cwff_\beta(\Sigma)$ with $\mathbf{A} \|^{\xi} \mathbf{B}$ be given. Regardless of the type β, the definition of $\|^{\xi}$ implies $\neg(\mathbf{A} \doteq^\beta \mathbf{B}){\downarrow_\beta} \notin \mathcal{H}$. So, we can apply the induction hypothesis to the two terms $h\mathbf{C}^1 \cdots \mathbf{C}^n\mathbf{A}$ and $h\mathbf{D}^1 \cdots \mathbf{D}^n\mathbf{B}$ at type γ to obtain $(h\mathbf{C}^1 \cdots \mathbf{C}^n\mathbf{A}) \|^{\xi} (h\mathbf{D}^1 \cdots \mathbf{D}^n\mathbf{B})$.

$\square$

Definition 2.4.29 (Decomposes Booleans) We say a Hintikka set $\mathcal{H}$ *decomposes booleans* if it satisfies $\vec{\nabla}_{\mathfrak{b}}$ or $\vec{\nabla}^h_{dec}$ for logical constants h.

We can now show reflexivity of $\|^{\mathfrak{f}}$ by induction on terms.

Lemma 2.4.30 *Let $\mathcal{H} \in \mathfrak{I}_\Sigma$ be a Hintikka set satisfying $\vec{\nabla}_{\mathfrak{f}}$, $\vec{\nabla}_m$ and $\vec{\nabla}^w_{dec}$ for every parameter $w \in \Sigma$. Furthermore, suppose that $\mathcal{H}$ decomposes booleans. Then, for every closed term $\mathbf{A}_\alpha$, $\mathbf{A} \|^{\mathfrak{f}} \mathbf{A}$.*

Proof: We prove the stronger statement that given any term $\mathbf{A}_\alpha \in \mathit{wff}_\alpha$ and substitutions θ and ψ defined on the free variables of $\mathbf{A}$ such that that $\theta(x_\beta) \|^{\mathfrak{f}} \psi(x_\beta)$ for every $x_\beta \in \mathit{dom}(\theta)$. We prove this by induction on the term $\mathbf{A}$.

If $\mathbf{A}$ is a variable, this follows from the assumption on θ and ψ. If $\mathbf{A}$ is a parameter, this follows by applying Lemma 2.4.27:2 to determine $\mathbf{A}$ is $\mathcal{H}$-compatible with itself. Lemma 2.4.27:2 also implies $\mathbf{A} \|^{\mathfrak{f}} \mathbf{A}$ if $\mathbf{A}$ is a logical constant and $\mathcal{H}$ satisfies $\vec{\nabla}^h_{dec}$ for logical constants h. Since $\mathcal{H}$ decomposes booleans, we either have $\vec{\nabla}^h_{dec}$ for logical constants h or $\vec{\nabla}_{\mathfrak{b}}$. So let us turn to the logical constants when $\mathcal{H}$ satisfies $\vec{\nabla}_{\mathfrak{b}}$.

The most interesting logical constants are the Π^β for each type β. If Π^β is $\mathcal{H}$-incompatible with itself, there must be two $\mathcal{H}$-compatible closed terms $\mathbf{B}, \mathbf{C} \in \mathit{cwff}_{\beta \to o}(\Sigma)$, such that $\Pi\mathbf{B}$ and $\Pi\mathbf{C}$ are $\mathcal{H}$-incompatible. By Lemma 2.4.25 (using $\vec{\nabla}_{\mathfrak{b}}$), this means without loss of generality $\Pi\mathbf{B}{\downarrow_\beta}, \neg(\Pi\mathbf{C}{\downarrow_\beta}) \in \mathcal{H}$. By $\overline{\nabla}_\exists$, there is a parameter w_β with $\neg\mathbf{C}w \in \mathcal{H}$. By $\overline{\nabla}_\forall$, $\mathbf{B}w \in \mathcal{H}$. By $\overline{\nabla}_\beta$, $(\mathbf{B}w){\downarrow_\beta}, \neg(\mathbf{C}w){\downarrow_\beta} \in \mathcal{H}$. We must have $\mathbf{B}w$ and $\mathbf{C}w$ are $\mathcal{H}$-incompatible by definition. Since we know $w \|^{\mathfrak{f}} w$ for parameter w already, this contradicts $\mathbf{B} \|^{\mathfrak{f}} \mathbf{C}$.

To check $\neg$, suppose $\neg$ is not $\mathcal{H}$-compatible with itself. There must be $\mathcal{H}$-compatible $\mathbf{B}, \mathbf{C} \in \mathit{cwff}_o(\Sigma)$ where $\neg\mathbf{B}$ and $\neg\mathbf{C}$ are $\mathcal{H}$-incompatible. Without loss of generality, using $\vec{\nabla}_{\mathfrak{b}}$ and Lemma 2.4.25, we have $\{\neg\mathbf{B}{\downarrow_\beta}, \neg\neg\mathbf{C}{\downarrow_\beta}\} \subseteq \mathcal{H}$. Using $\overline{\nabla}_\neg$, we have $\{\neg\mathbf{B}{\downarrow_\beta}, \mathbf{C}{\downarrow_\beta}\} \subseteq \mathcal{H}$, contradicting $\mathbf{B} \|^{\mathfrak{f}} \mathbf{C}$.

To check $\vee$, suppose $\vee$ is not $\mathcal{H}$-compatible with itself. We must have $\mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{E} \in \mathit{cwff}_o(\Sigma)$ with $\mathbf{B} \|^{\mathfrak{f}} \mathbf{C}$, $\mathbf{D} \|^{\mathfrak{f}} \mathbf{E}$ such that $\mathbf{B} \vee \mathbf{D}$ and $\mathbf{C} \vee \mathbf{E}$ are $\mathcal{H}$-incompatible. Without loss of generality, again using $\vec{\nabla}_{\mathfrak{b}}$ and Lemma 2.4.25, $\{(\mathbf{B} \vee \mathbf{D}){\downarrow_\beta}, \neg(\mathbf{C} \vee \mathbf{E}){\downarrow_\beta}\} \subseteq \mathcal{H}$. By $\overline{\nabla}_\wedge$ and $\overline{\nabla}_\vee$, either $\{\mathbf{B}{\downarrow_\beta}, \neg\mathbf{C}{\downarrow_\beta}, \neg\mathbf{E}{\downarrow_\beta})\} \subseteq \mathcal{H}$ (contradicting $\mathbf{B} \|^{\mathfrak{f}} \mathbf{C}$) or $\{\mathbf{D}{\downarrow_\beta}, \neg\mathbf{C}{\downarrow_\beta}, \neg\mathbf{E}{\downarrow_\beta})\} \subseteq \mathcal{H}$ (contradicting $\mathbf{D} \|^{\mathfrak{f}} \mathbf{E}$). $\neg((\mathbf{B} \vee \mathbf{D}) \doteq^o (\mathbf{C} \vee \mathbf{E})){\downarrow_\beta} \in \mathcal{H}$. So, $\vee \|^{\mathfrak{f}} \vee$.

For the application case let $\mathbf{A}$ be of the form $\mathbf{GB}$. By induction, we have $\theta(\mathbf{G}) \|^{\mathfrak{f}} \psi(\mathbf{G})$ and $\theta(\mathbf{B}) \|^{\mathfrak{f}} \psi(\mathbf{B})$. By definition of $\mathcal{H}$-compatibility at function types, we have $\theta(\mathbf{GB}) \equiv \theta(\mathbf{G})\theta(\mathbf{B}) \|^{\mathfrak{f}} \psi(\mathbf{G})\psi(\mathbf{B}) \equiv \psi(\mathbf{GB})$.

Suppose $\mathbf{A}$ is of the form $\lambda X_\beta.\mathbf{D}_\gamma$.[25] This case is where we need the stronger induction hypothesis. Let $\mathbf{B}, \mathbf{C} \in \mathit{cwff}_\beta(\Sigma)$ be $\mathcal{H}$-compatible terms. We need to show $\theta(\mathbf{A})\mathbf{B}$ and $\psi(\mathbf{A})\mathbf{C}$ are $\mathcal{H}$-compatible. Let $\theta' := \theta, [\mathbf{B}/X]$ and $\psi' := \psi, [\mathbf{C}/X]$ be the obvious substitutions extending θ and ψ. By induction, we have $\theta'(\mathbf{D}) \|^{\mathfrak{f}} \psi'(\mathbf{D})$. Since $\theta'(\mathbf{D}){\downarrow_\beta} \equiv (\theta(\mathbf{A})\mathbf{B}){\downarrow_\beta}$ and $\psi'(\mathbf{D}){\downarrow_\beta} \equiv (\psi(\mathbf{A})\mathbf{C}){\downarrow_\beta}$, we know by Lemma 2.4.24 that $\theta(\mathbf{A})\mathbf{B} \|^{\mathfrak{f}} \psi(\mathbf{A})\mathbf{C}$. Generalizing over $\mathbf{B}$ and $\mathbf{C}$, we know $\theta(\mathbf{A}) \|^{\mathfrak{f}} \psi(\mathbf{A})$. EdNote(25)

□

Similarly, we show reflexivity of $\|_{\xi}$.

Lemma 2.4.31 *Let $\mathcal{H} \in \mathfrak{I}_\Sigma$ be a Hintikka set satisfying $\vec{\nabla}_\xi$, $\vec{\nabla}_m$, $\vec{\nabla}_{\xi dec}$ and $\vec{\nabla}^w_{dec}$ for every parameter $w \in \Sigma$. Furthermore, suppose that $\mathcal{H}$ decomposes booleans. Then, for every closed term $\mathbf{A}_\alpha$, $\mathbf{A} \|_{\xi} \mathbf{A}$.*

Proof: We again prove the stronger statement that given any term $\mathbf{A}_\alpha \in \mathit{wff}_\alpha$ and substitutions θ and ψ defined on the free variables of $\mathbf{A}$ such that that $\theta(x_\beta) \|_{\xi} \psi(x_\beta)$ for every $x_\beta \in \mathit{dom}(\theta)$. We prove this by induction on the term $\mathbf{A}$.

[25] EdNote: this case made it necessary to distinguish "functional" from "boolean" compatibility

If $\mathbf{A}$ is a variable, this follows from the assumption on θ and ψ. If $\mathbf{A}$ is a parameter, this follows by applying Lemma 2.4.28 to determine $\mathbf{A}$ is $\mathcal{H}$-compatible with itself. Lemma 2.4.28 also implies $\mathbf{A} \|_\xi \mathbf{A}$ if $\mathbf{A}$ is a logical constant and $\mathcal{H}$ satisfies $\vec{\nabla}^h_{dec}$ for logical constants h. Since $\mathcal{H}$ decomposes booleans, we either have $\vec{\nabla}^h_{dec}$ for logical constants h or $\vec{\nabla}_{\mathfrak{b}}$. So let us turn to the logical constants when $\mathcal{H}$ satisfies $\vec{\nabla}_{\mathfrak{b}}$.

The most interesting logical constants are the Π^β for each type β. We know we cannot have $\neg(\Pi^\beta \doteq^{(\beta\to o)\to o} \Pi^\beta) \in \mathcal{H}$ by $\vec{\nabla}^r_{\doteq}$ (cf. Lemma **??**). So, if Π^β is $\mathcal{H}$-incompatible with itself, there must be two $\mathcal{H}$-compatible closed terms $\mathbf{B}, \mathbf{C} \in \mathit{cwff}_{\beta\to o}(\Sigma)$, such that $\Pi\mathbf{B}$ and $\Pi\mathbf{C}$ are $\mathcal{H}$-incompatible. By Lemma 2.4.25 (using $\vec{\nabla}_{\mathfrak{b}}$), this means without loss of generality $\Pi\mathbf{B}{\downarrow_\beta}, \neg(\Pi\mathbf{C}{\downarrow_\beta}) \in \mathcal{H}$. By $\overline{\nabla}_\exists$, there is a parameter w_β with $\neg\mathbf{C}w \in \mathcal{H}$. By $\overline{\nabla}_\forall$, $\mathbf{B}w \in \mathcal{H}$. By $\overline{\nabla}_\beta$, $(\mathbf{B}w){\downarrow_\beta}, \neg(\mathbf{C}w){\downarrow_\beta} \in \mathcal{H}$. We must have $\mathbf{B}w$ and $\mathbf{C}w$ are $\mathcal{H}$-incompatible by definition. Since we know $w \|_\xi w$ for parameter w already, this contradicts $\mathbf{B} \|_\xi \mathbf{C}$.

To check $\neg$, suppose $\neg$ is not $\mathcal{H}$-compatible with itself. There must be $\mathcal{H}$-compatible $\mathbf{B}, \mathbf{C} \in \mathit{cwff}_o(\Sigma)$ where $\neg\mathbf{B}$ and $\neg\mathbf{C}$ are $\mathcal{H}$-incompatible. Without loss of generality, using $\vec{\nabla}_{\mathfrak{b}}$ and Lemma 2.4.25, we have $\{\neg\mathbf{B}{\downarrow_\beta}, \neg\neg\mathbf{C}{\downarrow_\beta}\} \subseteq \mathcal{H}$. Using $\overline{\nabla}_\neg$, we have $\{\neg\mathbf{B}{\downarrow_\beta}, \mathbf{C}{\downarrow_\beta}\} \subseteq \mathcal{H}$, contradicting $\mathbf{B} \|_\xi \mathbf{C}$.

To check $\lor$, suppose $\lor$ is not $\mathcal{H}$-compatible with itself. We must have $\mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{E} \in \mathit{cwff}_o(\Sigma)$ with $\mathbf{B} \|_\xi \mathbf{C}$, $\mathbf{D} \|_\xi \mathbf{E}$ such that $\mathbf{B} \lor \mathbf{D}$ and $\mathbf{C} \lor \mathbf{E}$ are $\mathcal{H}$-incompatible. Without loss of generality, again using $\vec{\nabla}_{\mathfrak{b}}$ and Lemma 2.4.25, $\{(\mathbf{B} \lor \mathbf{D}){\downarrow_\beta}, \neg(\mathbf{C} \lor \mathbf{E}){\downarrow_\beta}\} \subseteq \mathcal{H}$. By $\overline{\nabla}_\land$ and $\overline{\nabla}_\lor$, either $\{\mathbf{B}{\downarrow_\beta}, \neg\mathbf{C}{\downarrow_\beta}, \neg\mathbf{E}{\downarrow_\beta})\} \subseteq \mathcal{H}$ (contradicting $\mathbf{B} \|_\xi \mathbf{C}$) or $\{\mathbf{D}{\downarrow_\beta}, \neg\mathbf{C}{\downarrow_\beta}, \neg\mathbf{E}{\downarrow_\beta})\} \subseteq \mathcal{H}$ (contradicting $\mathbf{D} \|_\xi \mathbf{E}$). $\neg((\mathbf{B} \lor \mathbf{D}) \doteq^o (\mathbf{C} \lor \mathbf{E})){\downarrow_\beta} \in \mathcal{H}$. So, $\lor \|_\xi \lor$.

For the application case let $\mathbf{A}$ be of the form $\mathbf{GB}$. By induction, we have $\theta(\mathbf{G}) \|_\xi \psi(\mathbf{G})$ and $\theta(\mathbf{B}) \|_\xi \psi(\mathbf{B})$. By definition of $\mathcal{H}$-ξ-compatibility at function types, we have $\theta(\mathbf{GB}) \equiv \theta(\mathbf{G})\theta(\mathbf{B}) \|_\xi \psi(\mathbf{G})\psi(\mathbf{B}) \equiv \psi(\mathbf{GB})$.

Suppose $\mathbf{A}$ is of the form $\lambda X_\beta.\mathbf{D}_\gamma$. First, we need to check that $\neg(\theta(\mathbf{A}) \doteq \psi(\mathbf{A})) \notin \mathcal{H}$. Note that $\theta(\mathbf{A})$ and $\psi(\mathbf{A})$ are both λ-abstractions. So, if $\neg(\theta(\mathbf{A}) \doteq \psi(\mathbf{A})) \in \mathcal{H}$, then by $\vec{\nabla}_\xi$ there is some parameter w_β such that $\neg((\theta, [w/X])(\mathbf{D}) \doteq (\psi, [w/X])(\mathbf{D})) \in \mathcal{H}$. Since we already know $w \|_\xi w$, the induction hypothesis implies $(\theta, [w/X])(\mathbf{D}) \|_\xi (\psi, [w/X])(\mathbf{D})$. By definition of ξ-compatibility, we must have $\neg(\theta, [w/X])(\mathbf{D}) \doteq (\psi, [w/X])(\mathbf{D})) \notin \mathcal{H}$, a contradiction.

Let $\mathbf{B}, \mathbf{C} \in \mathit{cwff}_\beta(\Sigma)$ be $\mathcal{H}$-compatible terms. We need to show $\theta(\mathbf{A})\mathbf{B}$ and $\psi(\mathbf{A})\mathbf{C}$ are $\mathcal{H}$-compatible. Let $\theta' := \theta, [\mathbf{B}/X]$ and $\psi' := \psi, [\mathbf{C}/X]$ be the obvious substitutions extending θ and ψ. By induction, we have $\theta'(\mathbf{D}) \|_\xi \psi'(\mathbf{D})$. Since $\theta'(\mathbf{D}){\downarrow_\beta} \equiv (\theta(\mathbf{A})\mathbf{B}){\downarrow_\beta}$ and $\psi'(\mathbf{D}){\downarrow_\beta} \equiv (\psi(\mathbf{A})\mathbf{C}){\downarrow_\beta}$, we know by Lemma 2.4.24 that $\theta(\mathbf{A})\mathbf{B} \|_\xi \psi(\mathbf{A})\mathbf{C}$. Generalizing over $\mathbf{B}$ and $\mathbf{C}$, we know $\theta(\mathbf{A}) \|_\xi \psi(\mathbf{A})$. $\Box$

It will be useful to record that the logical constants operate well on functionally and ξ-compatible sets. We first define some notation.

β : For $A \subseteq \mathit{cwff}_\alpha(\Sigma)$, let $A^\beta := \{\mathbf{A}{\downarrow_\beta} | \mathbf{A} \in A\}$.

$\neg$: For $A \subseteq \mathit{cwff}_o(\Sigma)$, let $\neg A := \{\neg\mathbf{A} | \mathbf{A} \in A\}$.

$\lor$: For $A \subseteq \mathit{cwff}_o(\Sigma)$ and $B \subseteq \mathit{cwff}_o(\Sigma)$, let $A \lor B := \{\mathbf{A} \lor \mathbf{B} | \mathbf{A} \in A, \mathbf{B} \in B\}$.

Π^α : For $F \subseteq \mathit{cwff}_{\alpha\to\beta}(\Sigma)$, let $\Pi^\alpha F := \{\Pi^\alpha\mathbf{F} | \mathbf{F} \in F\}$.

$\doteq^\alpha$: For $A \subseteq \mathit{cwff}_\alpha(\Sigma)$ and $B \subseteq \mathit{cwff}_\alpha(\Sigma)$, let $A \doteq^\alpha B := \{\mathbf{A} \doteq^\alpha \mathbf{B} | \mathbf{A} \in A, \mathbf{B} \in B\}$.

Lemma 2.4.32 *Let $\mathcal{H}$ be a Hintikka set satisfying $\vec{\nabla}_{\mathfrak{f}}$, $\vec{\nabla}_m$ and $\vec{\nabla}^w_{dec}$ for every parameter $w \in \Sigma$. Furthermore, suppose that $\mathcal{H}$ decomposes booleans.*

β**:** *If $A \subseteq cwff_\alpha(\Sigma)$ is $\mathcal{H}$-functionally compatible, then so is A^β.*

$\neg$**:** *If $A \subseteq cwff_o(\Sigma)$ is $\mathcal{H}$-functionally compatible, then so is $\neg A$.*

$\vee$**:** *If $A \subseteq cwff_o(\Sigma)$ and $B \subseteq cwff_o(\Sigma)$ are $\mathcal{H}$-functionally compatible, then so is $A \vee B$.*

Π^α**:** *If $F \subseteq cwff_{\alpha\to\beta}(\Sigma)$ is $\mathcal{H}$-functionally compatible, then so is $\Pi^\alpha F$.*

$\doteq^\alpha$**:** *If $A \subseteq cwff_\alpha(\Sigma)$ and $B \subseteq cwff_\alpha(\Sigma)$ are $\mathcal{H}$-functionally compatible, then so is $A \doteq^\alpha B$.*

Proof: The β case follows directly from Lemma 2.4.24. We can easily prove the other cases by applying Lemma 2.4.30. For example, Lemma 2.4.30 implies $\neg \mathrel{\|^{\mathfrak{f}}} \neg$. By the definition of $\mathrel{\|^{\mathfrak{f}}}$ at function types, we have $\neg\mathbf{A} \mathrel{\|^{\mathfrak{f}}} \neg\mathbf{B}$ whenever $\mathbf{A} \mathrel{\|^{\mathfrak{f}}} \mathbf{B}$. This verifies $\neg A$ is compatible whenever A is. Similarly, $\vee \mathrel{\|^{\mathfrak{f}}} \vee$, $\Pi^\alpha \mathrel{\|^{\mathfrak{f}}} \Pi^\alpha$ and $\doteq^\alpha \mathrel{\|^{\mathfrak{f}}} \doteq^\alpha$ verify the other cases. $\square$

Lemma 2.4.33 *Let $\mathcal{H}$ be a Hintikka set satisfying $\vec{\nabla}_\xi$, $\vec{\nabla}_m$, $\vec{\nabla}_{\xi dec}$ and $\vec{\nabla}^w_{dec}$ for every parameter $w \in \Sigma$. Furthermore, suppose that $\mathcal{H}$ decomposes booleans.*

β**:** *If $A \subseteq cwff_\alpha(\Sigma)$ is $\mathcal{H}$-ξ-compatible, then so is A^β.*

$\neg$**:** *If $A \subseteq cwff_o(\Sigma)$ is $\mathcal{H}$-ξ-compatible, then so is $\neg A$.*

$\vee$**:** *If $A \subseteq cwff_o(\Sigma)$ and $B \subseteq cwff_o(\Sigma)$ are $\mathcal{H}$-ξ-compatible, then so is $A \vee B$.*

Π^α**:** *If $F \subseteq cwff_{\alpha\to\beta}(\Sigma)$ is $\mathcal{H}$-ξ-compatible, then so is $\Pi^\alpha F$.*

$\doteq^\alpha$**:** *If $A \subseteq cwff_\alpha(\Sigma)$ and $B \subseteq cwff_\alpha(\Sigma)$ are $\mathcal{H}$-ξ-compatible, then so is $A \doteq^\alpha B$.*

Proof: The β case follows directly from Lemma 2.4.24. We can easily prove the other cases by applying Lemma 2.4.31. For example, Lemma 2.4.31 implies $\neg \mathrel{\|_{\xi}} \neg$. By the definition of $\mathrel{\|_{\xi}}$ at function types, we have $\neg\mathbf{A} \mathrel{\|_{\xi}} \neg\mathbf{B}$ whenever $\mathbf{A} \mathrel{\|_{\xi}} \mathbf{B}$. This verifies $\neg A$ is compatible whenever A is. Similarly, $\vee \mathrel{\|_{\xi}} \vee$, $\Pi^\alpha \mathrel{\|_{\xi}} \Pi^\alpha$ and $\doteq^\alpha \mathrel{\|_{\xi}} \doteq^\alpha$ verify the other cases. $\square$

2.4.8 Model Existence for $\mathfrak{Acc}_{\beta\mathfrak{f}}$ and $\mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$

Since we have constructed models for the least extensional cases, we now turn our attention immediately to the functionally extensional cases, including the fully extensional case. We construct functionally extensional models by putting a per evaluation over a possible values structure similar to the Andrews Structure. In the case where $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ is a Hintikka set in Γ_Σ, a class in $\mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$, we will construct Henkin models (i.e., satisfying $\mathfrak{f}$, $\mathfrak{b}$ and $\mathfrak{q}$).

In this section, we fix an acceptable abstract consistency class Γ_Σ in $\mathfrak{Acc}_{\beta\mathfrak{f}}$, and an acceptable abstract consistency class $\Gamma^{\mathfrak{b}}_\Sigma$ in $\mathfrak{Acc}_{\beta\mathfrak{f}\mathfrak{b}}$, a Hintikka set $\mathcal{H}_{\beta\mathfrak{f}}$ in Γ_Σ and a Hintikka set $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ in $\Gamma^{\mathfrak{b}}_\Sigma$. Note that $\mathcal{H}_{\beta\mathfrak{f}}$ and $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ satisfy $\vec{\nabla}_{\mathfrak{f}}$, $\vec{\nabla}_m$, and $\vec{\nabla}^w_{dec}$ for all parameters $w \in \Sigma$. Furthermore, $\mathcal{H}_{\beta\mathfrak{f}}$ satisfies $\vec{\nabla}^h_{dec}$ for logical constants h and $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ satisfies $\vec{\nabla}_{\mathfrak{b}}$, so both decompose booleans. We use $\mathcal{H}_{\mathfrak{f}*}$ to denote either $\mathcal{H}_{\beta\mathfrak{f}}$ or $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ in context, and use the notation $\mathrel{\|^{\mathfrak{f}}}$ in context to mean either the $\mathcal{H}_{\beta\mathfrak{f}}$-functional compatibility relation or the $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$-functional compatibility relation.

Definition 2.4.34 ($\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$-Compatibility Structure) We define a possible values structure we will call the *$\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$-Compatibility Structure* $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}}_{\mathfrak{f}\mathfrak{b}} := (\mathcal{D}, @)$ by

- Let $\mathcal{D}_o$ be the set of pairs $\langle \mathbf{A}_o, p\rangle$ where $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$ is β-normal and $p \in \mathcal{B}^{\mathbf{A}}_{\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}}$.
- Let $\mathcal{D}_\iota$ be the set of pairs $\langle \mathbf{A}_\iota, S\rangle$ where $\mathbf{A} \in \mathit{cwff}_\iota(\Sigma)$ is β-normal and $\mathbf{A} \in S$ and S is a $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$-functionally compatible subset of $\mathit{cwff}_\iota(\Sigma)$.
- Let $\mathcal{D}_{\alpha\to\beta}$ be the set of pairs $\langle \mathbf{G}, g\rangle$ where $\mathbf{G} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$ is β-normal and $g : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ such that for every $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$, $g(\langle \mathbf{A}, a\rangle) \equiv \langle \mathbf{B}, b\rangle$ implies $\mathbf{B}$ is the β-normal form of $\mathbf{GA}$.

We define @ by

$$\langle \mathbf{G}, g\rangle @\langle \mathbf{A}, a\rangle \equiv \langle \mathbf{B}, g(\langle \mathbf{A}, a\rangle)\rangle$$

where $\mathbf{B}$ is the β-normal form of $\mathbf{GA}$ for each $\langle \mathbf{G}, g\rangle \in \mathcal{D}_{\alpha\to\beta}$ and $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$.

A similar structure will be used to construct the per-evaluation model for $\mathcal{H}_{\beta\mathfrak{f}}$.

Definition 2.4.35 ($\mathcal{H}_{\beta\mathfrak{f}}$-Compatibility Structure) We define a possible values structure we will call the $\mathcal{H}_{\beta\mathfrak{f}}$-*Compatibility Structure* $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}}}_{\mathfrak{f}} := (\mathcal{D}, @)$ by

- Let $\mathcal{D}_o$ be the set of pairs $\langle \mathbf{A}_o, \langle S, p\rangle\rangle$ where $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$ is β-normal, S is a $\mathcal{H}_{\beta\mathfrak{f}}$-functionally compatible set, $\mathbf{A} \in S$, and $p \in \mathcal{B}^{\mathbf{A}}_{\mathcal{H}_{\beta\mathfrak{f}}}$.
- Let $\mathcal{D}_\iota$ be the set of pairs $\langle \mathbf{A}_\iota, S\rangle$ where $\mathbf{A} \in \mathit{cwff}_\iota(\Sigma)$ is β-normal and $\mathbf{A} \in S$ and S is a $\mathcal{H}_{\beta\mathfrak{f}}$-functionally compatible subset of $\mathit{cwff}_\iota(\Sigma)$.
- Let $\mathcal{D}_{\alpha\to\beta}$ be the set of pairs $\langle \mathbf{G}, g\rangle$ where $\mathbf{G} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$ is β-normal and $g : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ such that for every $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$, $g(\langle \mathbf{A}, a\rangle) \equiv \langle \mathbf{B}, b\rangle$ implies $\mathbf{B}$ is the β-normal form of $\mathbf{GA}$.

We define @ by

$$\langle \mathbf{G}, g\rangle @\langle \mathbf{A}, a\rangle \equiv \langle (\mathbf{GA}){\downarrow_\beta}, g(\langle \mathbf{A}, a\rangle)\rangle$$

for each $\langle \mathbf{G}, g\rangle \in \mathcal{D}_{\alpha\to\beta}$ and $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$.

Lemma 2.4.36 (Compatibility Structures) *Both $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}}}_{\mathfrak{f}}$ and $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}}_{\mathfrak{f}\mathfrak{b}}$ are possible values structures.*

Proof: It is trivial to check most of the conditions that $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}}}_{\mathfrak{f}}$ and $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}}_{\mathfrak{f}\mathfrak{b}}$ are possible values structures. To show there is an a for every $\mathbf{A} \in \mathit{cwff}_\iota(\Sigma)$ with $\langle \mathbf{A}{\downarrow_\beta}, a\rangle \in \mathcal{D}_\iota$, we can use Lemma 2.4.30 to show $\langle \mathbf{A}{\downarrow_\beta}, \{\mathbf{A}{\downarrow_\beta}\}\rangle \in \mathcal{D}_\iota$. For every $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$, we know $\mathcal{B}^{\mathbf{A}\downarrow_\beta}_{\mathcal{H}_{\beta\mathfrak{f}}}$ and $\mathcal{B}^{\mathbf{A}\downarrow_\beta}_{\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}}$ are nonempty. In $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}}_{\mathfrak{f}\mathfrak{b}}$, we have $\langle \mathbf{A}{\downarrow_\beta}, \mathsf{p}\rangle$ for $\mathsf{p} \in \mathcal{B}^{\mathbf{A}\downarrow_\beta}_{\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}}$. In $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}}}_{\mathfrak{f}}$, we can apply Lemma 2.4.30 to show $\langle \mathbf{A}, \langle\{\mathbf{A}\}, \mathsf{p}\rangle\rangle \in \mathcal{D}_o$ in $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}}}_{\mathfrak{f}}$ for $\mathsf{p} \in \mathcal{B}^{\mathbf{A}\downarrow_\beta}_{\mathcal{H}_{\beta\mathfrak{f}}}$. $\Box$

Let $\sim$ the functional per extensions on $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}}_{\mathfrak{f}\mathfrak{b}}$ and $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{f}}}_{\mathfrak{f}}$ of the equivalence relations defined on base types as follows.

- In $\mathcal{D}_o$, $\langle \mathbf{A}_o, p\rangle \sim \langle \mathbf{B}_o, q\rangle$ if $p \equiv q$.
- In $\mathcal{D}_\iota$, $\langle \mathbf{A}_\iota, S\rangle \sim \langle \mathbf{B}_\iota, R\rangle$ if $S \equiv R$.

This definition on $\mathcal{D}_o$ is meaningful even though $\mathcal{D}_o$ is defined differently in the two cases.

By Definition 2.4.7, at function types, $\langle \mathbf{G}_{\alpha\to\beta}, g\rangle \sim \langle \mathbf{H}_{\alpha\to\beta}, h\rangle$ if for every $\langle \mathbf{A}_\alpha, a\rangle \sim \langle \mathbf{B}_\beta, b\rangle$ in $\mathcal{D}_\alpha$, we have $g(\langle \mathbf{A}_\alpha, a\rangle) \sim h(\langle \mathbf{B}_\alpha, b\rangle)$ in $\mathcal{D}_\beta$.

As in Definition 2.4.5, at each type α we let

$$\overline{\mathcal{D}_\alpha} := \{\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha | \langle \mathbf{A}, a\rangle \sim \langle \mathbf{A}, a\rangle\}$$

Combining this with the notation restricting the first components, we let $\overline{\mathcal{D}_\alpha^{\mathbf{A}}} := \mathcal{D}_\alpha^{\mathbf{A}} \cap \overline{\mathcal{D}_\alpha}$.

Since every $\mathbf{A} \in \mathit{cwff}_\alpha(\Sigma)$ has a possible value by Lemma 2.4.14, we can choose a particular one $r^{\mathbf{A}}$ for each $\mathbf{A}$. This will act as a default value when necessary.

Lemma 2.4.37 *For each type α, we have*

1. *If $\langle \mathbf{A}, a\rangle \sim \langle \mathbf{B}, b\rangle$ in $\mathcal{D}_\alpha$, then $\mathbf{A} \|^{\mathfrak{f}} \mathbf{B}$.*
2. *If $S \subseteq \mathit{cwff}_\alpha(\Sigma)$ is functionally compatible, then for every $\mathbf{A} \in S$, there is a possible value $p^{\mathbf{A}}$ for $\mathbf{A}$. Furthermore, $\langle \mathbf{A}\big\downarrow_\beta, p^{\mathbf{A}}\rangle \sim \langle \mathbf{B}\big\downarrow_\beta, p^{\mathbf{B}}\rangle$ for each $\mathbf{A}, \mathbf{B} \in S$.*

Proof: These two statements are proven by a mutual induction over the type α.

1. **o, $\mathcal{H}_{\beta\mathfrak{fb}}$:** $\langle \mathbf{A}, a\rangle \sim \langle \mathbf{B}, b\rangle$ implies $a \equiv b \in \{\mathtt{T}, \mathtt{F}\}$. If $\mathbf{A}$ and $\mathbf{B}$ were functionally incompatible, then without loss of generality $\mathbf{A}, \neg\mathbf{B} \in \mathcal{H}_{\beta\mathfrak{fb}}$. Since $\mathbf{A} \in \mathcal{H}_{\beta\mathfrak{fb}}$, a cannot be $\mathtt{F}$, and so $a \equiv \mathtt{T}$. Since $\neg\mathbf{B} \in \mathcal{H}_{\beta\mathfrak{fb}}$, b cannot be $\mathtt{T}$, and so $b \equiv \mathtt{F}$. Since $\mathtt{T} \not\equiv \mathtt{F}$, we have a contradiction.

   o, $\mathcal{H}_{\beta\mathfrak{f}}$: $\langle \mathbf{A}, \langle A, p\rangle\rangle \sim \langle \mathbf{B}, \langle B, q\rangle\rangle$ implies $A \equiv B$. So, $\mathbf{A} \|^{\mathfrak{f}} \mathbf{B}$ as members of the same $\mathcal{H}_{\beta\mathfrak{f}}$-functionally compatible set A.

   ι: $\langle \mathbf{A}, A\rangle \sim \langle \mathbf{B}, B\rangle$ implies $A \equiv B$. So, $\mathbf{A} \|^{\mathfrak{f}} \mathbf{B}$ as members of the functionally compatible set A.

   $\beta \to \gamma$: Suppose $\langle \mathbf{G}, g\rangle \sim \langle \mathbf{H}, h\rangle$. Let $\mathbf{A} \|^{\mathfrak{f}} \mathbf{B}$ in $\mathit{cwff}_\beta(\Sigma)$ be given. Applying the induction hypothesis for part 2 at type β to the set $S := \{\mathbf{A}, \mathbf{B}\}$, we obtain $p^{\mathbf{A}}$ and $p^{\mathbf{B}}$ with $\langle \mathbf{A}\big\downarrow_\beta, p^{\mathbf{A}}\rangle \sim \langle \mathbf{B}\big\downarrow_\beta, p^{\mathbf{B}}\rangle$. So, $g(\langle \mathbf{A}\big\downarrow_\beta, p^{\mathbf{A}}\rangle) \sim h(\langle \mathbf{B}\big\downarrow_\beta, p^{\mathbf{B}}\rangle)$. The first components of $g(\langle \mathbf{A}\big\downarrow_\beta, p^{\mathbf{A}}\rangle)$ and $h(\langle \mathbf{B}\big\downarrow_\beta, p^{\mathbf{B}}\rangle)$ are $(\mathbf{GA})\big\downarrow_\beta$ and $(\mathbf{HB})\big\downarrow_\beta$, resp. Applying the induction hypothesis for part 1 to these terms at type γ, we have $(\mathbf{GA})\big\downarrow_\beta \|^{\mathfrak{f}} (\mathbf{HB})\big\downarrow_\beta$. By Lemma 2.4.24, $(\mathbf{GA}) \|^{\mathfrak{f}} (\mathbf{HB})$. Generalizing over $\mathbf{A}$ and $\mathbf{B}$, we have $\mathbf{G} \|^{\mathfrak{f}} \mathbf{H}$.

2. **o, $\mathcal{H}_{\beta\mathfrak{fb}}$:** We must either be able to let $p^{\mathbf{A}} := \mathtt{T}$ for every $\mathbf{A} \in S$ or let $p^{\mathbf{A}} := \mathtt{F}$ for every $\mathbf{A} \in S$. If neither is the case, then by the definition of $\mathcal{D}_o$ there must be $\mathbf{A}, \mathbf{B} \in S$ with $\mathbf{A}\big\downarrow_\beta, \neg\mathbf{B}\big\downarrow_\beta \in \mathcal{H}_{\beta\mathfrak{fb}}$. But this contradicts $\mathcal{H}_{\beta\mathfrak{fb}}$-functional compatibility of S.

   o, $\mathcal{H}_{\beta\mathfrak{f}}$: The set S^β is functionally compatible by Lemma 2.4.32. We must either be able to let $p^{\mathbf{A}} := \langle S^\beta, \mathtt{T}\rangle$ for every $\mathbf{A} \in S$ or $p^{\mathbf{A}} := \langle S^\beta, \mathtt{F}\rangle$ for every $\mathbf{A} \in S$. If not, there must be $\mathbf{A}, \mathbf{B} \in S$ with $\mathbf{A}\big\downarrow_\beta, \neg\mathbf{B}\big\downarrow_\beta \in \mathcal{H}_{\beta\mathfrak{f}}$. But this contradicts $\mathcal{H}_{\beta\mathfrak{f}}$-compatibility of S.

   ι: Let $p^{\mathbf{A}} := S^\beta$ for each $\mathbf{A} \in S$. Functional compatibility of S^β follows from Lemma 2.4.32. By definition of $\mathcal{D}_\iota$, $\langle \mathbf{A}\big\downarrow_\beta, S^\beta\rangle \in \mathcal{D}_\iota$ So, we let $p^{\mathbf{A}} := S^\beta$ for each $\mathbf{A} \in S$.

   $\beta \to \gamma$: Suppose we are given the set $S \subseteq \mathit{cwff}_{\beta\to\gamma}(\Sigma)$.

   For $\langle \mathbf{B}, b\rangle \in \mathcal{D}_\beta \setminus \overline{\mathcal{D}_\beta}$ and $\mathbf{G} \in S$, we let $p^{\mathbf{GB}}$ be the default possible value $r^{\mathbf{GB}}$ for $\mathbf{GB}$.

For each $\langle \mathbf{B}, b\rangle \in \overline{\mathcal{D}_\beta}$, we choose a particular representative $\langle \mathbf{B}^\sim, b^\sim\rangle$ in the equivalence class of $\langle \mathbf{B}, b\rangle$ with respect to $\sim$. For a particular $\langle \mathbf{B}^\sim, b^\sim\rangle$, let

$$\mathcal{B} := \{\langle \mathbf{B}, b\rangle | \langle \mathbf{B}, b\rangle \sim \langle \mathbf{B}^\sim, b^\sim\rangle\}$$

and let

$$\mathcal{G}_\mathcal{B} := \{\mathbf{GB} | \mathbf{G} \in S, \langle \mathbf{B}, b\rangle \in \mathcal{B} \text{ for some } b\}$$

For each $\langle \mathbf{B}, b\rangle, \langle \mathbf{C}, c\rangle \in \mathcal{B}$, Applying the induction hypothesis for part 1 to $\langle \mathbf{B}, b\rangle \sim \langle \mathbf{C}, c\rangle$ at type β, we have $\mathbf{B} \|^{\mathfrak{f}} \mathbf{C}$. So, the set $\mathcal{G}_\mathcal{B}$ is functionally compatible since S is functionally compatible by the definition of $\|^{\mathfrak{f}}$ at function types.
By applying the induction hypothesis for part 2 to $\mathcal{G}_\mathcal{B}$ at type γ we obtain related possible values $p^{\mathbf{GB}}$ for each $\mathbf{GB} \in \mathcal{G}_\mathcal{B}$. This defines $p^{\mathbf{GB}}$ for each $\mathbf{G} \in S$ and $\mathbf{B} \in \overline{\mathcal{D}_\beta}$. Now, for each $\mathbf{G} \in S$, we can use the axiom of choice (at the meta level) to define a function $p^{\mathbf{G}} : \mathcal{D}_\beta \to \mathcal{D}_\gamma$ such that

$$p^{\mathbf{G}}(\langle \mathbf{B}, b\rangle) \equiv \langle (\mathbf{GB})\big\downarrow_\beta, p^{\mathbf{GB}}\rangle$$

This $p^{\mathbf{G}}$ does map into $\mathcal{D}_\gamma$ since each $p^{\mathbf{GB}}$ is a possible value for $\mathbf{GB}$. Note that the choices of $p^{\mathbf{GB}}$ imply the functions $p^{\mathbf{G}}$ are related as

$$p^{\mathbf{G}}(\langle \mathbf{B}, b\rangle) \equiv \langle (\mathbf{GB})\big\downarrow_\beta, p^{\mathbf{GB}}\rangle \sim \langle (\mathbf{HC})\big\downarrow_\beta, p^{\mathbf{HC}}\rangle \equiv p^{\mathbf{H}}(\langle \mathbf{C}, c\rangle)$$

whenever $\langle \mathbf{B}, b\rangle \sim \langle \mathbf{C}, c\rangle$, so that $\langle \mathbf{G}, p^{\mathbf{G}}\rangle \sim \langle \mathbf{H}, p^{\mathbf{H}}\rangle$ for each $\mathbf{G}, \mathbf{H} \in S$. In particular, if $\mathbf{G}$ and $\mathbf{H}$ are both the same $\mathbf{G} \in S$, this verifies $\langle \mathbf{G}, p^{\mathbf{G}}\rangle \sim \langle \mathbf{G}, p^{\mathbf{G}}\rangle$. So, the $p^{\mathbf{G}}$ functions are related possible values for the members of S.

$\square$

Theorem 2.4.38 *Every closed term $\mathbf{A} \in cwff_\alpha(\Sigma)$ has a possible value a with $\langle \mathbf{A}\big\downarrow_\beta, a\rangle \in \overline{\mathcal{D}_\alpha}$. That is, $\overline{\mathcal{D}_\alpha^{\mathbf{A}}}$ is nonempty.*

Proof: This follows simply by applying Lemma2.4.37:2 to the singleton set $\{\mathbf{A}\}$ since $\mathbf{A} \|^{\mathfrak{f}} \mathbf{A}$ by Lemma 2.4.30. $\square$

We can now use this construction to prove the following theorem.

Theorem 2.4.39 (Model Existence ($\mathfrak{Hint}_{\beta\mathfrak{fb}}$, $\mathfrak{Hint}_{\beta\mathfrak{f}}$)) *Let $\mathcal{H}_{\mathfrak{f}*}$ be an acceptable Hintikka set in $\mathfrak{Hint}_{\beta\mathfrak{fb}}$ [$\mathfrak{Hint}_{\beta\mathfrak{f}}$]. There exists a model $\mathcal{M}$ in $\mathfrak{M}_{\beta\mathfrak{fb}}$ [$\mathfrak{M}_{\beta\mathfrak{f}}$] such that $\mathcal{M} \models \mathcal{H}_{\mathfrak{f}*}$.*

Proof: First, if $\mathcal{H}_{\mathfrak{f}*}$ is not equation-free, then it is complete by Theorem **??**. In this case, we are done by Theorem **??**. So, we may assume $\mathcal{H}_{\mathfrak{f}*}$ is equation-free.

Let $\mathcal{F} := \mathcal{F}_{\mathfrak{fb}}^{\mathcal{H}_{\beta\mathfrak{fb}}} \equiv (\mathcal{D}, @)$ [$\mathcal{F}_{\mathfrak{f}}^{\mathcal{H}_{\beta\mathfrak{f}}}$] be the $\mathcal{H}_{\beta\mathfrak{fb}}$-Compatibility Structure [$\mathcal{H}_{\beta\mathfrak{f}}$-Compatibility Structure]. This is a possible values structure by Lemma 2.4.36. To apply Theorem 2.4.17, we need to interpret the constants in Σ. For parameters $c_\alpha \in \Sigma$, then choose any possible value p^c with $\langle c, p^c\rangle \in \overline{\mathcal{D}_\alpha}$ for c. Such a possible value exists by Theorem 2.4.38. Let $\mathcal{I}(c) := \langle c, p^c\rangle$.

To interpret the logical constants, we must check that we can interpret $\neg$, $\vee$, and each Π^α for each α in the intended way. So, we define the appropriate function and then check that this is an appropriate possible value. Here we must distinguish between the $\mathcal{H}_{\beta\mathfrak{f}}$ and $\mathcal{H}_{\beta\mathfrak{fb}}$ cases.

To ease notation, we define a function $\upsilon : \mathcal{D}_o \to \{\mathtt{T}, \mathtt{F}\}$ by $\upsilon(\langle \mathbf{A}, \langle A, \mathsf{p}\rangle\rangle) := \mathsf{p}$ in the $\mathcal{H}_{\beta\mathfrak{f}}$ case and by $\upsilon(\langle \mathbf{A}, \mathsf{p}\rangle) := \mathsf{p}$ in the $\mathcal{H}_{\beta\mathfrak{fb}}$ case. Later, this will serve as the valuation in our per evaluation.

$\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$**:** Let $n : \mathcal{D}_o \to \mathcal{D}_o$ be the function taking $\langle \mathbf{A}, a\rangle$ to $\langle \neg\mathbf{A}, b\rangle$ where $b \equiv \mathtt{F}$ if $a \equiv \mathtt{T}$ and $b \equiv \mathtt{T}$ if $a \equiv \mathtt{F}$. It is easy to check using $\overline{\nabla_\neg}$ and ∇_c that $\langle \neg, n\rangle \in \mathcal{D}_{o\to o}$. It is also easy to check that $\langle \neg, n\rangle \in \overline{\mathcal{D}_{o\to o}}$ using the definition of $\sim$ at type o. Let $\mathcal{I}(\neg) := \langle \neg, n\rangle$.

For each $\langle \mathbf{A}, a\rangle \in \mathcal{D}_o$, let $d_{\langle \mathbf{A},a\rangle} : \mathcal{D}_o \to \mathcal{D}_o$ be defined by $d_{\langle \mathbf{A},a\rangle}(\langle \mathbf{B}, b\rangle) := \langle \mathbf{A} \vee \mathbf{B}, c\rangle$ where $c \equiv \mathtt{T}$ if $a \equiv \mathtt{T}$ or $b \equiv \mathtt{T}$, and $c \equiv \mathtt{F}$ otherwise. Using $\overline{\nabla_\wedge}$, $\overline{\nabla_\vee}$ and ∇_c and the definition of $\sim$ on $\mathcal{D}_o$, we can easily check that each $\langle \mathbf{A}, d_{\langle \mathbf{A},a\rangle}\rangle \in \overline{\mathcal{D}_{o\to o}}$. Furthermore, we can use these properties to show $\langle \mathbf{A}, d_{\langle \mathbf{A},a\rangle}\rangle \sim \langle \mathbf{A}', d_{\langle \mathbf{A}',a\rangle}\rangle$ for any other $\langle \mathbf{A}', a\rangle \in \mathcal{D}_o$. That is, $\sim$-related values in $\mathcal{D}_o$ give $\sim$-related values d_*. So, we let $d : \mathcal{D}_o \to \mathcal{D}_{o\to o}$ be defined by $d(\langle \mathbf{A}, a\rangle) := \langle \vee\mathbf{A}, d_{\langle \mathbf{A},a\rangle}\rangle$, and conclude $\langle \vee, d\rangle \in \overline{\mathcal{D}_{o\to o\to o}}$. Let $\mathcal{I}(\vee) := \langle \vee, d\rangle$.

The most interesting case is Π^α. Here we let $\langle \Pi^\alpha, \pi^\alpha\rangle$ where $\pi^\alpha : \mathcal{D}_{\alpha\to o} \to \mathcal{D}_o$ is defined by $\pi^\alpha(\langle \mathbf{F}, f\rangle) := \langle \Pi^\alpha\mathbf{F}, p\rangle$ where $p \equiv \mathtt{T}$ if $\upsilon(f(\mathsf{a})) \equiv \mathtt{T}$ for every $\mathsf{a} \in \overline{\mathcal{D}_\alpha}$ and $p \equiv \mathtt{F}$ otherwise. Note that we have relativized the Π^α quantifier to $\overline{\mathcal{D}_\alpha}$. In general, this will *not* give a Σ-model directly on the structure $\mathcal{F}$. But it will give an appropriate per evaluation on the evaluation we will build over $\mathcal{F}$.

We must check that $\langle \Pi^\alpha, \pi^\alpha\rangle \in \overline{\mathcal{D}_{(\alpha\to o)\to o}}$. First, $\overline{\nabla_\forall}$, $\overline{\nabla_\exists}$, $\overline{\nabla_\beta}$ and ∇_c (along with $\mathcal{I}(w) \sim \mathcal{I}(w)$ for parameters) imply π^α is well-defined and $\langle \Pi^\alpha, \pi^\alpha\rangle \in \mathcal{D}_{(\alpha\to o)\to o}$. To check $\langle \Pi^\alpha, \pi^\alpha\rangle \sim \langle \Pi^\alpha, \pi^\alpha\rangle$, let $\langle \mathbf{F}, f\rangle \sim \langle \mathbf{G}, g\rangle$ be given. By the definition of $\sim$, this implies $\upsilon(f(\mathsf{a})) \sim \upsilon(g(\mathsf{a}))$ for every $\mathsf{a} \in \overline{\mathcal{D}_\alpha}$. So, $\upsilon(\pi^\alpha(\langle \mathbf{F}, f\rangle)) \equiv \upsilon(\pi^\alpha(\langle \mathbf{G}, g\rangle))$ which precisely means $\pi^\alpha(\langle \mathbf{F}, f\rangle) \sim \pi^\alpha(\langle \mathbf{G}, g\rangle)$. So, let $\mathcal{I}(\Pi^\alpha) := \langle \Pi^\alpha, \pi^\alpha\rangle$.

$\mathcal{H}_{\beta\mathfrak{f}}$**:** Let $n : \mathcal{D}_o \to \mathcal{D}_o$ be the function taking $\langle \mathbf{A}, \langle A, a\rangle\rangle$ to $\langle \neg\mathbf{A}, \langle \neg A, b\rangle\rangle$ where $b \equiv \mathtt{F}$ if $a \equiv \mathtt{T}$ and $b \equiv \mathtt{T}$ if $a \equiv \mathtt{F}$. Note that $\neg A$ is compatible by Lemma 2.4.32. It is easy to check using $\overline{\nabla_\neg}$ and ∇_c that $\langle \neg, n\rangle \in \mathcal{D}_{o\to o}$. It is also easy to check that $\langle \neg, n\rangle \in \overline{\mathcal{D}_{o\to o}}$ using the definition of $\sim$ at type o. Let $\mathcal{I}(\neg) := \langle \neg, n\rangle$.

For each $\langle \mathbf{A}, \langle A, a\rangle\rangle \in \mathcal{D}_o$, let $d_{\langle \mathbf{A},\langle A,a\rangle\rangle} : \mathcal{D}_o \to \mathcal{D}_o$ be defined by $d_{\langle \mathbf{A},\langle A,a\rangle\rangle}(\langle \mathbf{B}, \langle B, b\rangle\rangle) := \langle \mathbf{A} \vee \mathbf{B}, \langle A \vee B, c\rangle\rangle$ where $c \equiv \mathtt{T}$ if $a \equiv \mathtt{T}$ or $b \equiv \mathtt{T}$, and $c \equiv \mathtt{F}$ otherwise. Note that $A \vee B$ is compatible by Lemma 2.4.32. Using $\overline{\nabla_\wedge}$, $\overline{\nabla_\vee}$ and ∇_c and the definition of $\sim$ on $\mathcal{D}_o$, we can easily check that each $\langle \mathbf{A}, d_{\langle \mathbf{A},\langle A,a\rangle\rangle}\rangle \in \overline{\mathcal{D}_{o\to o}}$. Furthermore, we can use these properties to show $\langle \mathbf{A}, d_{\langle \mathbf{A},\langle A,a\rangle\rangle}\rangle \sim \langle \mathbf{A}', d_{\langle \mathbf{A}',\langle A,a\rangle\rangle}\rangle$ for any other $\langle \mathbf{A}', a\rangle \in \mathcal{D}_o$. That is, $\sim$-related values in $\mathcal{D}_o$ give $\sim$-related values d_*. So, we let $d : \mathcal{D}_o \to \mathcal{D}_{o\to o}$ be defined by $d(\langle \mathbf{A}, \langle A, a\rangle\rangle) := \langle \vee\mathbf{A}, d_{\langle \mathbf{A},\langle A,a\rangle\rangle}\rangle$, and conclude $\langle \vee, d\rangle \in \overline{\mathcal{D}_{o\to o\to o}}$. Let $\mathcal{I}(\vee) := \langle \vee, d\rangle$.

Again, we let $\langle \Pi^\alpha, \pi^\alpha\rangle$ where $\pi^\alpha : \mathcal{D}_{\alpha\to o} \to \mathcal{D}_o$. This time, we must define π^α more carefully. Let $Z_{\langle \mathbf{F},f\rangle} := \{\mathbf{G} | \exists g. \langle \mathbf{G}, g\rangle \sim \langle \mathbf{F}, f\rangle\}$. This set $Z_{\langle \mathbf{F},f\rangle}$ is compatible by Lemma 2.4.37:1. The set $\Pi^\alpha Z_{\langle \mathbf{F},f\rangle}$ is compatible by Lemma 2.4.32. Define $\pi^\alpha(\langle \mathbf{F}, f\rangle) := \langle \Pi^\alpha\mathbf{F}, \langle \Pi^\alpha Z_{\langle \mathbf{F},f\rangle}, p\rangle\rangle$ where $p \equiv \mathtt{T}$ if $\upsilon(f(\mathsf{a})) \equiv \mathtt{T}$ for every $\mathsf{a} \in \overline{\mathcal{D}_\alpha}$ and $p \equiv \mathtt{F}$ otherwise. Again, we have relativized the Π^α quantifier to $\overline{\mathcal{D}_\alpha}$.

We must check that $\langle \Pi^\alpha, \pi^\alpha\rangle \in \overline{\mathcal{D}_{(\alpha\to o)\to o}}$. First, $\overline{\nabla_\forall}$, $\overline{\nabla_\exists}$, $\overline{\nabla_\beta}$ and ∇_c (along with $\mathcal{I}(w) \sim \mathcal{I}(w)$) imply π^α is well-defined and $\langle \Pi^\alpha, \pi^\alpha\rangle \in \mathcal{D}_{(\alpha\to o)\to o}$. To check $\langle \Pi^\alpha, \pi^\alpha\rangle \sim \langle \Pi^\alpha, \pi^\alpha\rangle$, let $\langle \mathbf{F}, f\rangle \sim \langle \mathbf{G}, g\rangle$ be given. By the definition of $\sim$, this implies $\upsilon(f(\mathsf{a})) \sim \upsilon(g(\mathsf{a}))$ for every $\mathsf{a} \in \overline{\mathcal{D}_\alpha}$. So, $\upsilon(\pi^\alpha(\langle \mathbf{F}, f\rangle)) \equiv \upsilon(\pi^\alpha(\langle \mathbf{G}, g\rangle))$. To conclude $\pi^\alpha(\langle \mathbf{F}, f\rangle) \sim \pi^\alpha(\langle \mathbf{G}, g\rangle)$, we still need to show $\Pi^\alpha Z_{\langle \mathbf{F},f\rangle} \equiv \Pi^\alpha Z_{\langle \mathbf{G},g\rangle}$. This follows directly from $Z_{\langle \mathbf{F},f\rangle} \equiv Z_{\langle \mathbf{G},g\rangle}$ which always holds for $\langle \mathbf{F}, f\rangle \sim \langle \mathbf{G}, g\rangle$. So, let $\mathcal{I}(\Pi^\alpha) := \langle \Pi^\alpha, \pi^\alpha\rangle$.

Now we have an interpretation function $\mathcal{I}$ such that the first component of each $\mathcal{I}(c)$ is c for every $c \in \Sigma$. Furthermore, $\mathcal{I}(c) \sim \mathcal{I}(c)$ for every $c \in \Sigma$, so by Theorem 2.4.17, there is an evaluation function $\mathcal{E}$ such that

1. $\mathcal{E}|_{\Sigma} \equiv \mathcal{I}$,

2. $\mathcal{A} := (\mathcal{D}, @, \mathcal{E})$ is a possible values evaluation, hence a Σ-evaluation, and

3. $\mathcal{E}_\varphi(\mathbf{A}) \in \mathcal{D}_\alpha^{\theta^\varphi(\mathbf{A})}$ for each $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$.

This last condition implies $\upsilon(\mathcal{E}(\mathbf{A})) \equiv \mathtt{T}$ for each $\mathbf{A} \in \mathcal{H}_{\mathfrak{f}*}$, since $\mathcal{B}^{\mathbf{A}}_{\mathcal{H}_{\mathfrak{f}*}} \equiv \{\mathtt{T}\}$.

We now show $(\mathcal{A}, \sim, \upsilon)$ is a Σ-per evaluation satisfying $\partial^{o,\iota}$, ∂^{Σ}, $\partial^{\mathfrak{f}}$, ∂^{υ}, $\partial^{\neg}$, $\partial^{\vee}$, ∂^{Π} and $\partial^{\mathfrak{q}}$. Then we will apply Theorem 2.4.10. We already know $\mathcal{A}$ is a Σ-evaluation and $\sim$ is a typed per on the domains $\mathcal{D}_\alpha$. The only other condition to check is that υ is surjective. By ∇_c, $\overline{\nabla_\neg}$ and $\overline{\nabla_\beta}$, there must be a normal $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$ with $\neg\mathbf{A} \notin \mathcal{H}_{\mathfrak{f}*}$. So, in the $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ case $\langle \mathbf{A}, \mathtt{T}\rangle, \langle \neg\mathbf{A}, \mathtt{F}\rangle \in \mathcal{D}_o$. Let $\mathsf{p} := \langle \mathbf{A}, \mathtt{T}\rangle$ and $\mathsf{q} := \langle \neg\mathbf{A}, \mathtt{F}\rangle$. In the $\mathcal{H}_{\beta\mathfrak{f}}$ case $\langle \mathbf{A}, \langle\{\mathbf{A}\}, \mathtt{T}\rangle\rangle, \langle \neg\mathbf{A}, \langle\{\neg\mathbf{A}\}, \mathtt{F}\rangle\rangle \in \mathcal{D}_o$. Let $\mathsf{p} := \langle \mathbf{A}, \langle\{\mathbf{A}\}, \mathtt{T}\rangle\rangle$ and $\mathsf{q} := \langle \neg\mathbf{A}, \langle\{\neg\mathbf{A}\}, \mathtt{F}\rangle\rangle$. Now, in either case, $\upsilon(\mathsf{p}) \equiv \mathtt{T}$ and $\upsilon(\mathsf{q}) \equiv \mathtt{F}$, and υ is surjective.

So, $\mathcal{P} := (\mathcal{A}, \sim, \upsilon)$ is a Σ-per evaluation. We now check $\mathcal{P}$ has each of the desired properties.

$\partial^{o,\iota}$: On base types, $\sim$ is a total equivalence relation by definition.

∂^{Σ}: For each constant $c \in \Sigma$, $\mathcal{E}(c) \equiv \mathcal{I}(c) \sim \mathcal{I}(c) \equiv \mathcal{E}(c)$.

∂^{c}, $\partial^{\mathfrak{f}}$, ∂^{s}: By the definition of $\sim$ on function domains $\mathcal{D}_{\alpha\to\beta}$, for each $\mathsf{g}, \mathsf{h} \in \mathcal{D}_{\alpha\to\beta}$, $\mathsf{g} \sim \mathsf{h}$ iff for every $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$ $\mathsf{a} \sim \mathsf{b}$ implies $\mathsf{g}@\mathsf{a} \sim \mathsf{h}@\mathsf{b}$. By Lemma 2.4.9, we have ∂^{s}.

∂^{υ}: We have $\mathsf{a} \sim \mathsf{b}$ implies $\upsilon(\mathsf{a}) \equiv \upsilon(\mathsf{b})$ for each $\mathsf{a}, \mathsf{b} \in \mathcal{D}_o$ by the definition of $\sim$ at type o.

$\partial^{\neg}$: This follows from the definition of $n : \mathcal{D}_o \to \mathcal{D}_o$ above.

$\partial^{\vee}$: This follows from the definition of d above.

∂^{Π}: This follows from the definition of π^α (which was relativized to the smaller domains $\overline{\mathcal{D}_\alpha}$) above.

$\partial^{\mathfrak{q}}$: This is the only property that is difficult to check. We will use the fact that $\mathcal{H}_{\mathfrak{f}*}$ is equation-free. Let a type α be given. We need to find an element $\mathrm{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that for all $\mathsf{a}, \mathsf{b} \in \overline{\mathcal{D}_\alpha^\sim}$, $\upsilon(\mathrm{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$ iff $\mathsf{a} \sim \mathsf{b}$. We distinguish between the $\mathcal{H}_{\beta\mathfrak{f}}$ and $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ cases.

$\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$: For each $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$, let $s_\mathsf{a} : \mathcal{D}_\alpha \to \mathcal{D}_o$ be defined by $s_\mathsf{a}(\mathsf{b}) := \langle (\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \mathtt{T}\rangle$ for $\mathsf{b} \equiv \langle \mathbf{B}, b\rangle$ if $\mathsf{b} \sim \mathsf{a}$ in $\mathcal{D}_\alpha$, and $s_\mathsf{a}(\mathsf{b}) := \langle (\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \mathtt{F}\rangle$ otherwise. To show this is well-defined, we need to know $s_\mathsf{a}(\mathsf{b})$ really is in $\mathcal{D}_o$. If $\mathsf{a} \sim \mathsf{b}$ and $s_\mathsf{a}(\mathsf{b}) \equiv ((\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \mathtt{T}) \notin \mathcal{D}_o$, then we must have $\neg(\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta} \in \mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$. This contradicts $\mathsf{a} \sim \mathsf{b}$ by Lemma 2.4.37:1. If $\mathsf{a} \not\sim \mathsf{b}$ and $s_\mathsf{a}(\mathsf{b}) \equiv ((\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \mathtt{F}) \notin \mathcal{D}_o$, then we must have $(\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta} \in \mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$, contradicting the assumption that $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ is equation-free.[26] Note that this includes the case where $\mathsf{a} \not\sim \mathsf{a}$. So we now have $\langle (\lambda X_\alpha . \mathbf{A} \doteq^\alpha x){\downarrow_\beta}, s_\mathsf{a}\rangle \in \mathcal{D}_{\alpha\to o}$. EdNote(26)
Now, define $l : \mathcal{D}_\alpha \to \mathcal{D}_{\alpha\to o}$ by $l(\mathsf{a}) := \langle (\lambda X_\alpha . \mathbf{A} \doteq^\alpha x){\downarrow_\beta}, s_\mathsf{a}\rangle$ for any $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$. Since each $s_\mathsf{a} \in \mathcal{D}_{\alpha\to o}$, this is well-defined. Let $\mathrm{q}^\alpha := \langle \doteq^\alpha, l\rangle$. We must check $\mathrm{q}^\alpha \sim \mathrm{q}^\alpha$. Let $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$ and $\mathsf{b} \equiv \langle \mathbf{B}, b\rangle \in \mathcal{D}_\alpha$ with $\mathsf{a} \sim \mathsf{b}$ be given. We need to check $l(\mathsf{a}) \sim l(\mathsf{b})$. That is, we must check that s_a and s_b sends $\sim$-related values to $\sim$-related results. Let $\mathsf{c} \equiv \langle \mathbf{C}, c\rangle \in \mathcal{D}_\alpha$ and $\mathsf{d} \equiv \langle \mathbf{D}, d\rangle \in \mathcal{D}_\alpha$ with $\mathsf{c} \sim \mathsf{d}$ be given. $\upsilon(s_\mathsf{a}(\mathsf{c})) \equiv \mathtt{T}$ iff $\mathsf{a} \sim \mathsf{c}$ iff $\mathsf{b} \sim \mathsf{d}$ iff $\upsilon(s_\mathsf{b}(\mathsf{d})) \equiv \mathtt{T}$. So, $s_\mathsf{a}(\mathsf{c}) \sim s_\mathsf{b}(\mathsf{d})$.

[26] EdNote: this is where we need $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ is equation-free

$\mathcal{H}_{\beta\mathfrak{f}}$: Since $\mathcal{D}_o$ was defined differently in this case, we must define each s_{a} more carefully. For each $\mathsf{a} \in \mathcal{D}_\alpha$, define $S_{\mathsf{a}} := \{\mathbf{C} | \exists c. \langle \mathbf{C}, c\rangle \sim \mathsf{a}\}$. This is compatible by Lemma 2.4.37:1. For any $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$, $S_{\mathsf{a}} \doteq^\alpha S_{\mathsf{b}}$ is compatible by Lemma 2.4.32.

Let $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$ be given. For each $\mathsf{b} \equiv \langle \mathbf{B}, b\rangle \in \mathcal{D}_\alpha$, define $s_{\mathsf{a}}(\mathsf{b}) := \langle (\mathbf{A} \doteq^\alpha \mathbf{B})\downarrow_\beta, \langle S_{\mathsf{a}} \doteq^\alpha S_{\mathsf{b}}, \mathtt{T}\rangle\rangle$ if $\mathsf{a} \sim \mathsf{b}$, and $s_{\mathsf{a}}(\mathsf{b}) := \langle (\mathbf{A} \doteq^\alpha \mathbf{B})\downarrow_\beta, \langle S_{\mathsf{a}} \doteq^\alpha S_{\mathsf{b}}, \mathtt{F}\rangle\rangle$ otherwise. The verification that s_{a} is well-defined is just as in the $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ case. As before, we define $\mathrm{q}^\alpha := \langle \doteq^\alpha, l\rangle$ where $l : \mathcal{D}_\alpha \to \mathcal{D}_{\alpha\to o}$ is defined by $l(\mathsf{a}) := \langle (\lambda X_\alpha . \mathbf{A} \doteq^\alpha x)\downarrow_\beta, s_{\mathsf{a}}\rangle$ for any $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$. The verification that $\mathrm{q}^\alpha \sim \mathrm{q}^\alpha$ again reduces to showing that if $\mathsf{a} \sim \mathsf{b}$ and $\mathsf{c} \sim \mathsf{d}$, then $s_{\mathsf{a}}(\mathsf{c}) \sim s_{\mathsf{b}}(\mathsf{d})$, but this time we have more to check to verify $\sim$ at type o.

Let $\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d} \in \mathcal{D}_\alpha$ with $\langle \mathbf{A}, a\rangle \equiv \mathsf{a} \sim \mathsf{b} \equiv \langle \mathbf{B}, b\rangle$, $\langle \mathbf{C}, c\rangle \equiv \mathsf{c} \sim \mathsf{d} \equiv \langle \mathbf{D}, d\rangle$ be given. By definition, we have $s_{\mathsf{a}}(\mathsf{c}) \equiv \langle (\mathbf{A} \doteq^\alpha \mathbf{C})\downarrow_\beta, \langle S_{\mathsf{a}} \doteq^\alpha S_{\mathsf{c}}, p\rangle\rangle$ and $s_{\mathsf{b}}(\mathsf{d}) \equiv \langle (\mathbf{B} \doteq^\alpha \mathbf{D})\downarrow_\beta, \langle S_{\mathsf{b}} \doteq^\alpha S_{\mathsf{d}}, q\rangle\rangle$. We can easily show $p \equiv q$ as before, since $p \equiv \mathtt{T}$ iff $\mathsf{a} \sim \mathsf{c}$ iff $\mathsf{b} \sim \mathsf{d}$ iff $q \equiv \mathtt{T}$. It remains to show $(S_{\mathsf{a}} \doteq^\alpha S_{\mathsf{c}}) \equiv (S_{\mathsf{b}} \doteq^\alpha S_{\mathsf{d}})$. This follows from $S_{\mathsf{a}} \equiv S_{\mathsf{b}}$ (since $\mathsf{a} \sim \mathsf{b}$) and $S_{\mathsf{c}} \equiv S_{\mathsf{d}}$ (since $\mathsf{c} \sim \mathsf{d}$).

This verifies $\partial^{\mathfrak{q}}$ in both cases.

Furthermore, in the $\mathcal{H}_{\beta\mathfrak{f}\mathfrak{b}}$ case, we clearly have only two $\sim$-equivalence classes in $\mathcal{D}_o$ since there are only two possibilities for the second component of each $\mathsf{a} \in \mathcal{D}_o$. So, we have $\partial^{\mathfrak{b}}$ in this case.

We can now apply Theorem 2.4.10 to obtain a model $\mathcal{M} \equiv (\mathcal{D}^\sim, @^\sim, \mathcal{E}^\sim, \upsilon^\sim)$ in $\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$ [$\mathfrak{M}_{\beta\mathfrak{f}}$] such that $\upsilon^\sim(\mathcal{E}^\sim(\mathbf{A})) \equiv \upsilon(\mathcal{E}(\mathbf{A}))$ for all $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$. Note that $\mathcal{M} \models \mathcal{H}_{\mathfrak{f}*}$ since for each $\mathbf{A} \in \mathcal{H}_{\mathfrak{f}*}$ $\upsilon^\sim(\mathcal{E}^\sim(\mathbf{A})) \equiv \upsilon(\mathcal{E}(\mathbf{A})) \equiv \mathtt{T}$. $\square$

2.4.9 Model Existence for $\mathfrak{Acc}_{\beta\xi}$ and $\mathfrak{Acc}_{\beta\xi\mathfrak{b}}$

We now turn to the cases for ξ.

Definition 2.4.40 ($\mathcal{H}_{\beta\xi\mathfrak{b}}$-Compatibility Structure) We define a possible values structure we will call the *$\mathcal{H}_{\beta\xi\mathfrak{b}}$-Compatibility Structure* $\mathcal{F}_{\xi\mathfrak{b}}^{\mathcal{H}_{\beta\xi\mathfrak{b}}} := (\mathcal{D}, @)$ by

- Let $\mathcal{D}_o$ be the set of pairs $\langle \mathbf{A}_o, p\rangle$ where $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$ is β-normal and $p \in \mathcal{B}^{\mathbf{A}}_{\mathcal{H}_{\beta\xi\mathfrak{b}}}$.
- Let $\mathcal{D}_\iota$ be the set of pairs $\langle \mathbf{A}_\iota, S\rangle$ where $\mathbf{A} \in \mathit{cwff}_\iota(\Sigma)$ is β-normal and $\mathbf{A} \in S$ and S is a $\mathcal{H}_{\beta\xi\mathfrak{b}}$-$\xi$-compatible subset of $\mathit{cwff}_\iota(\Sigma)$.
- Let $\mathcal{D}_{\alpha\to\beta}$ be the set of pairs $\langle \mathbf{G}, g\rangle$ where $\mathbf{G} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$ is β-normal and $g : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ such that for every $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$, $g(\langle \mathbf{A}, a\rangle) \equiv \langle \mathbf{B}, b\rangle$ implies $\mathbf{B}$ is the β-normal form of $\mathbf{GA}$.

We define @ by

$$\langle \mathbf{G}, g\rangle @ \langle \mathbf{A}, a\rangle \equiv \langle \mathbf{B}, g(\langle \mathbf{A}, a\rangle)\rangle$$

where $\mathbf{B}$ is the β-normal form of $\mathbf{GA}$ for each $\langle \mathbf{G}, g\rangle \in \mathcal{D}_{\alpha\to\beta}$ and $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$.

A similar structure will be used to construct the per-evaluation model for $\mathcal{H}_{\beta\xi}$.

Definition 2.4.41 ($\mathcal{H}_{\beta\xi}$-Compatibility Structure) We define a possible values structure we will call the *$\mathcal{H}_{\beta\xi}$-Compatibility Structure* $\mathcal{F}_{\xi}^{\mathcal{H}_{\beta\xi}} := (\mathcal{D}, @)$ by

- Let $\mathcal{D}_o$ be the set of pairs $\langle \mathbf{A}_o, \langle S, p\rangle\rangle$ where $\mathbf{A} \in cwff_o(\Sigma)$ is β-normal, S is a $\mathcal{H}_{\beta\mathfrak{\xi}}$-$\xi$-compatible set, $\mathbf{A} \in S$, and $p \in \mathcal{B}^{\mathbf{A}}_{\mathcal{H}_{\beta\mathfrak{\xi}}}$.
- Let $\mathcal{D}_\iota$ be the set of pairs $\langle \mathbf{A}_\iota, S\rangle$ where $\mathbf{A} \in cwff_\iota(\Sigma)$ is β-normal and $\mathbf{A} \in S$ and S is a $\mathcal{H}_{\beta\mathfrak{\xi}}$-$\xi$-compatible subset of $cwff_\iota(\Sigma)$.
- Let $\mathcal{D}_{\alpha\to\beta}$ be the set of pairs $\langle \mathbf{G}, g\rangle$ where $\mathbf{G} \in cwff_{\alpha\to\beta}(\Sigma)$ is β-normal and $g : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ such that for every $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$, $g(\langle \mathbf{A}, a\rangle) \equiv \langle \mathbf{B}, b\rangle$ implies $\mathbf{B}$ is the β-normal form of $\mathbf{GA}$.

We define @ by

$$\langle \mathbf{G}, g\rangle @\langle \mathbf{A}, a\rangle \equiv \langle (\mathbf{GA})\big\downarrow_\beta, g(\langle \mathbf{A}, a\rangle)\rangle$$

for each $\langle \mathbf{G}, g\rangle \in \mathcal{D}_{\alpha\to\beta}$ and $\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$.

Lemma 2.4.42 (Compatibility Structures) *Both $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{\xi}}}_{\xi}$ and $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{\xi}\mathfrak{b}}}_{\xi\mathfrak{b}}$ are possible values structures.*

Proof: It is trivial to check most of the conditions that $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{\xi}}}_{\xi}$ and $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{\xi}\mathfrak{b}}}_{\xi\mathfrak{b}}$ are possible values structures. To show there is an a for every $\mathbf{A} \in cwff_\iota(\Sigma)$ with $\langle \mathbf{A}\big\downarrow_\beta, a\rangle \in \mathcal{D}_\iota$, we can use Lemma 2.4.31 to show $\langle \mathbf{A}\big\downarrow_\beta, \{\mathbf{A}\big\downarrow_\beta\}\rangle \in \mathcal{D}_\iota$. For every $\mathbf{A} \in cwff_o(\Sigma)$, we know $\mathcal{B}^{\mathbf{A}\downarrow_\beta}_{\mathcal{H}_{\beta\mathfrak{\xi}}}$ and $\mathcal{B}^{\mathbf{A}\downarrow_\beta}_{\mathcal{H}_{\beta\mathfrak{\xi}\mathfrak{b}}}$ are nonempty. In $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{\xi}\mathfrak{b}}}_{\xi\mathfrak{b}}$, we have $\langle \mathbf{A}\big\downarrow_\beta, \mathsf{p}\rangle$ for $\mathsf{p} \in \mathcal{B}^{\mathbf{A}\downarrow_\beta}_{\mathcal{H}_{\beta\mathfrak{\xi}\mathfrak{b}}}$. In $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{\xi}}}_{\xi}$, we can apply Lemma 2.4.31 to show $\langle \mathbf{A}, \langle \{\mathbf{A}\}, \mathsf{p}\rangle\rangle \in \mathcal{D}_o$ in $\mathcal{F}^{\mathcal{H}_{\beta\mathfrak{\xi}}}_{\xi}$ for $\mathsf{p} \in \mathcal{B}^{\mathbf{A}\downarrow_\beta}_{\mathcal{H}_{\beta\mathfrak{\xi}}}$. $\square$

In this case, we cannot use the functional per extension since we cannot expect to construct a functional model in general. Instead, we define $\sim$ by induction on types as follows.

- In $\mathcal{D}_o$, $\langle \mathbf{A}_o, p\rangle \sim \langle \mathbf{B}_o, q\rangle$ if $p \equiv q$.
- In $\mathcal{D}_\iota$, $\langle \mathbf{A}_\iota, S\rangle \sim \langle \mathbf{B}_\iota, R\rangle$ if $S \equiv R$.
- In $\mathcal{D}_{\beta\to\gamma}$, $\langle \mathbf{F}_{\beta\to\gamma}, f\rangle \sim \langle \mathbf{G}_{\beta\to\gamma}, g\rangle$ if $\neg(\mathbf{F} \doteq \mathbf{G}) \notin \mathcal{H}_{\beta\mathfrak{\xi}\mathfrak{b}}$ $[\neg(\mathbf{F} \doteq \mathbf{G}) \notin \mathcal{H}_{\beta\mathfrak{\xi}}]$ and for all $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\beta$ with $\mathsf{a} \sim \mathsf{b}$, we have $f(\mathsf{a}) \sim g(\mathsf{b})$.

Again, this definition on $\mathcal{D}_o$ is meaningful even though $\mathcal{D}_o$ is defined differently in the two cases.

As in Definition 2.4.5, at each type α we let

$$\overline{\mathcal{D}_\alpha} := \{\langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha | \langle \mathbf{A}, a\rangle \sim \langle \mathbf{A}, a\rangle\}$$

Combining this with the notation restricting the first components, we let $\overline{\mathcal{D}^{\mathbf{A}}_\alpha} := \mathcal{D}^{\mathbf{A}}_\alpha \cap \overline{\mathcal{D}_\alpha}$.

Since every $\mathbf{A} \in cwff_\alpha(\Sigma)$ has a possible value by Lemma 2.4.14, we can choose a particular one $r^{\mathbf{A}}$ for each $\mathbf{A}$. This will act as a default value when necessary.

Lemma 2.4.43 *For each type α, we have*

1. *If $\langle \mathbf{A}, a\rangle \sim \langle \mathbf{B}, b\rangle$ in $\mathcal{D}_\alpha$, then $\mathbf{A}\|_{\xi} \mathbf{B}$.*
2. *If $S \subseteq cwff_\alpha(\Sigma)$ is ξ-compatible, then for every $\mathbf{A} \in S$, there is a possible value $p^{\mathbf{A}}$ for $\mathbf{A}$. Furthermore, $\langle \mathbf{A}\big\downarrow_\beta, p^{\mathbf{A}}\rangle \sim \langle \mathbf{B}\big\downarrow_\beta, p^{\mathbf{B}}\rangle$ for each $\mathbf{A}, \mathbf{B} \in S$.*

Proof: These two statements are proven by a mutual induction over the type α.

1. o, $\mathcal{H}_{\beta\xi\mathfrak{b}}$**:** $\langle \mathbf{A}, a\rangle \sim \langle \mathbf{B}, b\rangle$ implies $a \equiv b \in \{\mathtt{T}, \mathtt{F}\}$. If $\mathbf{A}$ and $\mathbf{B}$ were ξ-incompatible, then without loss of generality $\mathbf{A}, \neg\mathbf{B} \in \mathcal{H}_{\beta\xi\mathfrak{b}}$. Since $\mathbf{A} \in \mathcal{H}_{\beta\xi\mathfrak{b}}$, a cannot be $\mathtt{F}$, and so $a \equiv \mathtt{T}$. Since $\neg\mathbf{B} \in \mathcal{H}_{\beta\xi\mathfrak{b}}$, b cannot be $\mathtt{T}$, and so $b \equiv \mathtt{F}$. Since $\mathtt{T} \not\equiv \mathtt{F}$, we have a contradiction.

   o, $\mathcal{H}_{\beta\xi}$**:** $\langle \mathbf{A}, \langle A, p\rangle\rangle \sim \langle \mathbf{B}, \langle B, q\rangle\rangle$ implies $A \equiv B$. So, $\mathbf{A} \|_{\xi} \mathbf{B}$ as members of the same $\mathcal{H}_{\beta\xi}$-ξ-compatible set A.

   ι**:** $\langle \mathbf{A}, A\rangle \sim \langle \mathbf{B}, B\rangle$ implies $A \equiv B$. So, $\mathbf{A} \|_{\xi} \mathbf{B}$ as members of the ξ-compatible set A.

   $\beta \to \gamma$**:** Suppose $\langle \mathbf{G}, g\rangle \sim \langle \mathbf{H}, h\rangle$. By the definition of $\sim$ on function types, we know $\neg(\mathbf{G} \doteq \mathbf{H}) \notin \mathcal{H}_{\beta\xi\mathfrak{b}}$ $[\neg(\mathbf{G} \doteq \mathbf{H}) \notin \mathcal{H}_{\beta\xi}]$. Let $\mathbf{A} \|_{\xi} \mathbf{B}$ in $\mathit{cwff}_{\beta}(\Sigma)$ be given. Applying the induction hypothesis for part 2 at type β to the set $S := \{\mathbf{A}, \mathbf{B}\}$, we obtain $p^{\mathbf{A}}$ and $p^{\mathbf{B}}$ with $\langle \mathbf{A}{\downarrow_\beta}, p^{\mathbf{A}}\rangle \sim \langle \mathbf{B}{\downarrow_\beta}, p^{\mathbf{B}}\rangle$. So, $g(\langle \mathbf{A}{\downarrow_\beta}, p^{\mathbf{A}}\rangle) \sim h(\langle \mathbf{B}{\downarrow_\beta}, p^{\mathbf{B}}\rangle)$. The first components of $g(\langle \mathbf{A}{\downarrow_\beta}, p^{\mathbf{A}}\rangle)$ and $h(\langle \mathbf{B}{\downarrow_\beta}, p^{\mathbf{B}}\rangle)$ are $(\mathbf{GA}){\downarrow_\beta}$ and $(\mathbf{HB}){\downarrow_\beta}$, resp. Applying the induction hypothesis for part 1 to these terms at type γ, we have $(\mathbf{GA}){\downarrow_\beta} \|_{\xi} (\mathbf{HB}){\downarrow_\beta}$. By Lemma 2.4.24, $(\mathbf{GA}) \|_{\xi} (\mathbf{HB})$. Generalizing over $\mathbf{A}$ and $\mathbf{B}$, we have $\mathbf{G} \|_{\xi} \mathbf{H}$.

2. o, $\mathcal{H}_{\beta\xi\mathfrak{b}}$**:** We must either be able to let $p^{\mathbf{A}} := \mathtt{T}$ for every $\mathbf{A} \in S$ or let $p^{\mathbf{A}} := \mathtt{F}$ for every $\mathbf{A} \in S$. If neither is the case, then by the definition of $\mathcal{D}_o$ there must be $\mathbf{A}, \mathbf{B} \in S$ with $\mathbf{A}{\downarrow_\beta}, \neg\mathbf{B}{\downarrow_\beta} \in \mathcal{H}_{\beta\xi\mathfrak{b}}$. But this contradicts $\mathcal{H}_{\beta\xi\mathfrak{b}}$-functional compatibility of S.

   o, $\mathcal{H}_{\beta\xi}$**:** The set S^β is ξ-compatible by Lemma 2.4.33. We must either be able to let $p^{\mathbf{A}} := \langle S^\beta, \mathtt{T}\rangle$ for every $\mathbf{A} \in S$ or $p^{\mathbf{A}} := \langle S^\beta, \mathtt{F}\rangle$ for every $\mathbf{A} \in S$. If not, there must be $\mathbf{A}, \mathbf{B} \in S$ with $\mathbf{A}{\downarrow_\beta}, \neg\mathbf{B}{\downarrow_\beta} \in \mathcal{H}_{\beta\xi}$. But this contradicts $\mathcal{H}_{\beta\xi}$-compatibility of S.

   ι**:** Let $p^{\mathbf{A}} := S^\beta$ for each $\mathbf{A} \in S$. ξ-compatibility of S^β follows from Lemma 2.4.33. By definition of $\mathcal{D}_\iota$, $\langle \mathbf{A}{\downarrow_\beta}, S^\beta\rangle \in \mathcal{D}_\iota$ So, we let $p^{\mathbf{A}} := S^\beta$ for each $\mathbf{A} \in S$.

   $\beta \to \gamma$**:** Suppose we are given the set $S \subseteq \mathit{cwff}_{\beta\to\gamma}(\Sigma)$.
   For $\langle \mathbf{B}, b\rangle \in \mathcal{D}_\beta \setminus \overline{\mathcal{D}_\beta}$ and $\mathbf{G} \in S$, we let $p^{\mathbf{GB}}$ be the default possible value $r^{\mathbf{GB}}$ for $\mathbf{GB}$. For each $\langle \mathbf{B}, b\rangle \in \overline{\mathcal{D}_\beta}$, we choose a particular representative $\langle \mathbf{B}^{\sim}, b^{\sim}\rangle$ in the equivalence class of $\langle \mathbf{B}, b\rangle$ with respect to $\sim$. For a particular $\langle \mathbf{B}^{\sim}, b^{\sim}\rangle$, let

   $$\mathcal{B} := \{\langle \mathbf{B}, b\rangle | \langle \mathbf{B}, b\rangle \sim \langle \mathbf{B}^{\sim}, b^{\sim}\rangle\}$$

   and let

   $$\mathcal{G}_{\mathcal{B}} := \{\mathbf{GB} | \mathbf{G} \in S, \langle \mathbf{B}, b\rangle \in \mathcal{B} \text{ for some } b\}$$

   For each $\langle \mathbf{B}, b\rangle, \langle \mathbf{C}, c\rangle \in \mathcal{B}$, Applying the induction hypothesis for part 1 to $\langle \mathbf{B}, b\rangle \sim \langle \mathbf{C}, c\rangle$ at type β, we have $\mathbf{B} \|_{\xi} \mathbf{C}$. So, the set $\mathcal{G}_{\mathcal{B}}$ is ξ-compatible since S is ξ-compatible by the definition of $\|_{\xi}$ at function types.
   By applying the induction hypothesis for part 2 to $\mathcal{G}_{\mathcal{B}}$ at type γ we obtain related possible values $p^{\mathbf{GB}}$ for each $\mathbf{GB} \in \mathcal{G}_{\mathcal{B}}$. This defines $p^{\mathbf{GB}}$ for each $\mathbf{G} \in S$ and $\mathbf{B} \in \overline{\mathcal{D}_\beta}$. Now, for each $\mathbf{G} \in S$, we can use the axiom of choice (at the metalevel) to define a function $p^{\mathbf{G}} : \mathcal{D}_\beta \to \mathcal{D}_\gamma$ such that

   $$p^{\mathbf{G}}(\langle \mathbf{B}, b\rangle) \equiv \langle (\mathbf{GB}){\downarrow_\beta}, p^{\mathbf{GB}}\rangle$$

   This $p^{\mathbf{G}}$ does map into $\mathcal{D}_\gamma$ since each $p^{\mathbf{GB}}$ is a possible value for $\mathbf{GB}$. Note that the choices of $p^{\mathbf{GB}}$ imply the functions $p^{\mathbf{G}}$ are related as

   $$p^{\mathbf{G}}(\langle \mathbf{B}, b\rangle) \equiv \langle (\mathbf{GB}){\downarrow_\beta}, p^{\mathbf{GB}}\rangle \sim \langle (\mathbf{HC}){\downarrow_\beta}, p^{\mathbf{HC}}\rangle \equiv p^{\mathbf{H}}(\langle \mathbf{C}, c\rangle)$$

whenever $\langle \mathbf{B}, b\rangle \sim \langle \mathbf{C}, c\rangle$ for each $\mathbf{G}, \mathbf{H} \in S$. Also, by the definition of ξ-compatibility, we know $\neg(\mathbf{G} \doteq \mathbf{H}) \notin \mathcal{H}_{\beta\xi\mathfrak{b}}$ [$\neg(\mathbf{G} \doteq \mathbf{H}) \notin \mathcal{H}_{\beta\xi}$] for each $\mathbf{G}, \mathbf{H} \in S$. So, $\langle \mathbf{G}, p^{\mathbf{G}}\rangle \sim \langle \mathbf{H}, p^{\mathbf{H}}\rangle$ for each $\mathbf{G}, \mathbf{H} \in S$. In particular, if $\mathbf{G}$ and $\mathbf{H}$ are both the same $\mathbf{G} \in S$, this verifies $\langle \mathbf{G}, p^{\mathbf{G}}\rangle \sim \langle \mathbf{G}, p^{\mathbf{G}}\rangle$. So, the $p^{\mathbf{G}}$ functions are related possible values for the members of S.

□

Theorem 2.4.44 *Every closed term $\mathbf{A} \in cwff_\alpha(\Sigma)$ has a possible value a with $\langle \mathbf{A}{\downarrow_\beta}, a\rangle \in \overline{\mathcal{D}_\alpha}$. That is, $\overline{\mathcal{D}_\alpha^{\mathbf{A}}}$ is nonempty.*

Proof: This follows simply by applying Lemma2.4.43:2 to the singleton set $\{\mathbf{A}\}$ since $\mathbf{A}\|_\xi \mathbf{A}$ by Lemma 2.4.31. □

We can now use this construction to prove the following theorem.

Theorem 2.4.45 (Model Existence ($\mathfrak{Hint}_{\beta\xi\mathfrak{b}}$, $\mathfrak{Hint}_{\beta\xi}$)) *Let $\mathcal{H}_{\xi*}$ be an acceptable Hintikka set in $\mathfrak{Hint}_{\beta\xi\mathfrak{b}}$ [$\mathfrak{Hint}_{\beta\xi}$]. There exists a model $\mathcal{M}$ in $\mathfrak{M}_{\beta\xi\mathfrak{b}}$ [$\mathfrak{M}_{\beta\xi}$] such that $\mathcal{M} \models \mathcal{H}_{\xi*}$.*

Proof: First, if $\mathcal{H}_{\xi*}$ is not equation-free, then it is complete by Theorem **??**. In this case, we are done by Theorem **??**. So, we may assume $\mathcal{H}_{\xi*}$ is equation-free.

Let $\mathcal{F} := \mathcal{F}_{\xi\mathfrak{b}}^{\mathcal{H}_{\beta\xi\mathfrak{b}}} \equiv (\mathcal{D}, @)$ [$\mathcal{F}_\xi^{\mathcal{H}_{\beta\xi}}$] be the $\mathcal{H}_{\beta\xi\mathfrak{b}}$-Compatibility Structure [$\mathcal{H}_{\beta\xi}$-Compatibility Structure]. This is a possible values structure by Lemma 2.4.36. To apply Theorem 2.4.17, we need to interpret the constants in Σ. For parameters $c_\alpha \in \Sigma$, then choose any possible value p^c with $\langle c, p^c\rangle \in \overline{\mathcal{D}_\alpha}$ for c. Such a possible value exists by Theorem 2.4.44. Let $\mathcal{I}(c) := \langle c, p^c\rangle$.

To interpret the logical constants, we must check that we can interpret $\neg$, $\vee$, and each Π^α for each α in the intended way. So, we define the appropriate function and then check that this is an appropriate possible value. Here we must distinguish between the $\mathcal{H}_{\beta\xi}$ and $\mathcal{H}_{\beta\xi\mathfrak{b}}$ cases.

To ease notation, we define a function $v : \mathcal{D}_o \to \{\mathtt{T}, \mathtt{F}\}$ by $v(\langle \mathbf{A}, \langle A, \mathsf{p}\rangle\rangle) := \mathsf{p}$ in the $\mathcal{H}_{\beta\xi}$ case and by $v(\langle \mathbf{A}, \mathsf{p}\rangle) := \mathsf{p}$ in the $\mathcal{H}_{\beta\xi\mathfrak{b}}$ case. Later, this will serve as the valuation in our per evaluation.

$\mathcal{H}_{\beta\xi\mathfrak{b}}$: Let $n : \mathcal{D}_o \to \mathcal{D}_o$ be the function taking $\langle \mathbf{A}, a\rangle$ to $\langle \neg\mathbf{A}, b\rangle$ where $b \equiv \mathtt{F}$ if $a \equiv \mathtt{T}$ and $b \equiv \mathtt{T}$ if $a \equiv \mathtt{F}$. It is easy to check using $\overline{\nabla_\neg}$ and ∇_c that $\langle \neg, n\rangle \in \mathcal{D}_{o\to o}$. It is also easy to check that $\langle \neg, n\rangle \in \overline{\mathcal{D}_{o\to o}}$ using the definition of $\sim$ at type o. Let $\mathcal{I}(\neg) := \langle \neg, n\rangle$.

For each $\langle \mathbf{A}, a\rangle \in \mathcal{D}_o$, let $d_{\langle \mathbf{A},a\rangle} : \mathcal{D}_o \to \mathcal{D}_o$ be defined by $d_{\langle \mathbf{A},a\rangle}(\langle \mathbf{B}, b\rangle) := \langle \mathbf{A} \vee \mathbf{B}, c\rangle$ where $c \equiv \mathtt{T}$ if $a \equiv \mathtt{T}$ or $b \equiv \mathtt{T}$, and $c \equiv \mathtt{F}$ otherwise. Using $\overline{\nabla_\wedge}$, $\overline{\nabla_\vee}$ and ∇_c and the definition of $\sim$ on $\mathcal{D}_o$, we can easily check that each $\langle \mathbf{A}, d_{\langle \mathbf{A},a\rangle}\rangle \in \overline{\mathcal{D}_{o\to o}}$. Furthermore, we can use these properties to show $\langle \mathbf{A}, d_{\langle \mathbf{A},a\rangle}\rangle \sim \langle \mathbf{A}', d_{\langle \mathbf{A}',a\rangle}\rangle$ for any other $\langle \mathbf{A}', a\rangle \in \mathcal{D}_o$. That is, $\sim$-related values in $\mathcal{D}_o$ give $\sim$-related values d_*. So, we let $d : \mathcal{D}_o \to \overline{\mathcal{D}_{o\to o}}$ be defined by $d(\langle \mathbf{A}, a\rangle) := \langle \vee\mathbf{A}, d_{\langle \mathbf{A},a\rangle}\rangle$, and conclude $\langle \vee, d\rangle \in \overline{\mathcal{D}_{o\to o\to o}}$. Let $\mathcal{I}(\vee) := \langle \vee, d\rangle$.

The most interesting case is Π^α. Here we let $\langle \Pi^\alpha, \pi^\alpha\rangle$ where $\pi^\alpha : \mathcal{D}_{\alpha\to o} \to \mathcal{D}_o$ is defined by $\pi^\alpha(\langle \mathbf{F}, f\rangle) := \langle \Pi^\alpha\mathbf{F}, p\rangle$ where $p \equiv \mathtt{T}$ if $v(f(\mathsf{a})) \equiv \mathtt{T}$ for every $\mathsf{a} \in \overline{\mathcal{D}_\alpha}$ and $p \equiv \mathtt{F}$ otherwise. Note that we have relativized the Π^α quantifier to $\overline{\mathcal{D}_\alpha}$. In general, this will *not* give a Σ-model directly on the structure $\mathcal{F}$. But it will give an appropriate per evaluation on the evaluation we will build over $\mathcal{F}$.

We must check that $\langle \Pi^\alpha, \pi^\alpha\rangle \in \overline{\mathcal{D}_{(\alpha\to o)\to o}}$. First, $\overline{\nabla_\forall}$, $\overline{\nabla_\exists}$, $\overline{\nabla_\beta}$ and ∇_c (along with $\mathcal{I}(w) \sim \mathcal{I}(w)$ for parameters) imply π^α is well-defined and $\langle \Pi^\alpha, \pi^\alpha\rangle \in \mathcal{D}_{(\alpha\to o)\to o}$. To check $\langle \Pi^\alpha, \pi^\alpha\rangle \sim \langle \Pi^\alpha, \pi^\alpha\rangle$, let $\langle \mathbf{F}, f\rangle \sim \langle \mathbf{G}, g\rangle$ be given. By the definition of $\sim$, this implies $v(f(\mathsf{a})) \sim v(g(\mathsf{a}))$

for every $\mathsf{a} \in \overline{\mathcal{D}_\alpha}$. So, $v(\pi^\alpha(\langle \mathbf{F}, f\rangle)) \equiv v(\pi^\alpha(\langle \mathbf{G}, g\rangle))$ which precisely means $\pi^\alpha(\langle \mathbf{F}, f\rangle) \sim \pi^\alpha(\langle \mathbf{G}, g\rangle)$. So, let $\mathcal{I}(\Pi^\alpha) := \langle \Pi^\alpha, \pi^\alpha\rangle$.

$\mathcal{H}_{\beta\xi}$: Let $n : \mathcal{D}_o \to \mathcal{D}_o$ be the function taking $\langle \mathbf{A}, \langle A, a\rangle\rangle$ to $\langle \neg\mathbf{A}, \langle \neg A, b\rangle\rangle$ where $b \equiv \mathtt{F}$ if $a \equiv \mathtt{T}$ and $b \equiv \mathtt{T}$ if $a \equiv \mathtt{F}$. Note that $\neg A$ is compatible by Lemma 2.4.33. It is easy to check using $\overline{\nabla}_\neg$ and ∇_c that $\langle \neg, n\rangle \in \mathcal{D}_{o\to o}$. It is also easy to check that $\langle \neg, n\rangle \in \overline{\mathcal{D}_{o\to o}}$ using the definition of $\sim$ at type o. Let $\mathcal{I}(\neg) := \langle \neg, n\rangle$.

For each $\langle \mathbf{A}, \langle A, a\rangle\rangle \in \mathcal{D}_o$, let $d_{\langle \mathbf{A}, \langle A, a\rangle\rangle} : \mathcal{D}_o \to \mathcal{D}_o$ be defined by $d_{\langle \mathbf{A}, \langle A, a\rangle\rangle}(\langle \mathbf{B}, \langle B, b\rangle\rangle) := \langle \mathbf{A} \vee \mathbf{B}, \langle A \vee B, c\rangle\rangle$ where $c \equiv \mathtt{T}$ if $a \equiv \mathtt{T}$ or $b \equiv \mathtt{T}$, and $c \equiv \mathtt{F}$ otherwise. Note that $A \vee B$ is compatible by Lemma 2.4.33. Using $\overline{\nabla}_\wedge$, $\overline{\nabla}_\vee$ and ∇_c and the definition of $\sim$ on $\mathcal{D}_o$, we can easily check that each $\langle \mathbf{A}, d_{\langle \mathbf{A}, \langle A, a\rangle\rangle}\rangle \in \overline{\mathcal{D}_{o\to o}}$. Furthermore, we can use these properties to show $\langle \mathbf{A}, d_{\langle \mathbf{A}, \langle A, a\rangle\rangle}\rangle \sim \langle \mathbf{A}', d_{\langle \mathbf{A}', \langle A, a\rangle\rangle}\rangle$ for any other $\langle \mathbf{A}', a\rangle \in \mathcal{D}_o$. That is, $\sim$-related values in $\mathcal{D}_o$ give $\sim$-related values d_*. So, we let $d : \mathcal{D}_o \to \mathcal{D}_{o\to o}$ be defined by $d(\langle \mathbf{A}, \langle A, a\rangle\rangle) := \langle \vee\mathbf{A}, d_{\langle \mathbf{A}, \langle A, a\rangle\rangle}\rangle$, and conclude $\langle \vee, d\rangle \in \overline{\mathcal{D}_{o\to o\to o}}$. Let $\mathcal{I}(\vee) := \langle \vee, d\rangle$.

Again, we let $\langle \Pi^\alpha, \pi^\alpha\rangle$ where $\pi^\alpha : \mathcal{D}_{\alpha\to o} \to \mathcal{D}_o$. This time, we must define π^α more carefully. Let $Z_{\langle \mathbf{F}, f\rangle} := \{\mathbf{G} | \exists g.\langle \mathbf{G}, g\rangle \sim \langle \mathbf{F}, f\rangle\}$. This set $Z_{\langle \mathbf{F}, f\rangle}$ is compatible by Lemma 2.4.43:1. The set $\Pi^\alpha Z_{\langle \mathbf{F}, f\rangle}$ is compatible by Lemma 2.4.33. Define $\pi^\alpha(\langle \mathbf{F}, f\rangle) := \langle \Pi^\alpha \mathbf{F}, \langle \Pi^\alpha Z_{\langle \mathbf{F}, f\rangle}, p\rangle\rangle$ where $p \equiv \mathtt{T}$ if $v(f(\mathsf{a})) \equiv \mathtt{T}$ for every $\mathsf{a} \in \overline{\mathcal{D}_\alpha}$ and $p \equiv \mathtt{F}$ otherwise. Again, we have relativized the Π^α quantifier to $\overline{\mathcal{D}_\alpha}$.

We must check that $\langle \Pi^\alpha, \pi^\alpha\rangle \in \overline{\mathcal{D}_{(\alpha\to o)\to o}}$. First, $\overline{\nabla}_\forall$, $\overline{\nabla}_\exists$, $\overline{\nabla}_\beta$ and ∇_c (along with $\mathcal{I}(w) \sim \mathcal{I}(w)$) imply π^α is well-defined and $\langle \Pi^\alpha, \pi^\alpha\rangle \in \mathcal{D}_{(\alpha\to o)\to o}$. To check $\langle \Pi^\alpha, \pi^\alpha\rangle \sim \langle \Pi^\alpha, \pi^\alpha\rangle$, let $\langle \mathbf{F}, f\rangle \sim \langle \mathbf{G}, g\rangle$ be given. By the definition of $\sim$, this implies $v(f(\mathsf{a})) \sim v(g(\mathsf{a}))$ for every $\mathsf{a} \in \overline{\mathcal{D}_\alpha}$. So, $v(\pi^\alpha(\langle \mathbf{F}, f\rangle)) \equiv v(\pi^\alpha(\langle \mathbf{G}, g\rangle))$. To conclude $\pi^\alpha(\langle \mathbf{F}, f\rangle) \sim \pi^\alpha(\langle \mathbf{G}, g\rangle)$, we still need to show $\Pi^\alpha Z_{\langle \mathbf{F}, f\rangle} \equiv \Pi^\alpha Z_{\langle \mathbf{G}, g\rangle}$. This follows directly from $Z_{\langle \mathbf{F}, f\rangle} \equiv Z_{\langle \mathbf{G}, g\rangle}$ which always holds for $\langle \mathbf{F}, f\rangle \sim \langle \mathbf{G}, g\rangle$. So, let $\mathcal{I}(\Pi^\alpha) := \langle \Pi^\alpha, \pi^\alpha\rangle$.

Now we have an interpretation function $\mathcal{I}$ such that the first component of each $\mathcal{I}(c)$ is c for every $c \in \Sigma$. Furthermore, $\mathcal{I}(c) \sim \mathcal{I}(c)$ for every $c \in \Sigma$, so by Theorem 2.4.17, there is an evaluation function $\mathcal{E}$ such that

1. $\mathcal{E}\big|_\Sigma \equiv \mathcal{I}$,
2. $\mathcal{A} := (\mathcal{D}, @, \mathcal{E})$ is a possible values evaluation, hence a Σ-evaluation, and
3. $\mathcal{E}_\varphi(\mathbf{A}) \in \mathcal{D}_\alpha^{\theta^\varphi(\mathbf{A})}$ for each $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$.

This last condition implies $v(\mathcal{E}(\mathbf{A})) \equiv \mathtt{T}$ for each $\mathbf{A} \in \mathcal{H}_{\xi*}$, since $\mathcal{B}^{\mathbf{A}}_{\mathcal{H}_{\xi*}} \equiv \{\mathtt{T}\}$.

We now show $(\mathcal{A}, \sim, v)$ is a Σ-per evaluation satisfying $\partial^{o,\iota}$, ∂^Σ, ∂^s, ∂^v, $\partial^\neg$, $\partial^\vee$, ∂^Π, ∂^ξ and $\partial^{\mathfrak{q}}$. Then we will apply Theorem 2.4.10. We already know $\mathcal{A}$ is a Σ-evaluation and $\sim$ is a typed per on the domains $\mathcal{D}_\alpha$. The only other condition to check is that v is surjective. By ∇_c, $\overline{\nabla}_\neg$ and $\overline{\nabla}_\beta$, there must be a normal $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$ with $\neg\mathbf{A} \notin \mathcal{H}_{\xi*}$. So, in the $\mathcal{H}_{\beta\xi\mathfrak{b}}$ case $\langle \mathbf{A}, \mathtt{T}\rangle, \langle \neg\mathbf{A}, \mathtt{F}\rangle \in \mathcal{D}_o$. Let $\mathsf{p} := \langle \mathbf{A}, \mathtt{T}\rangle$ and $\mathsf{q} := \langle \neg\mathbf{A}, \mathtt{F}\rangle$. In the $\mathcal{H}_{\beta\xi}$ case $\langle \mathbf{A}, \langle \{\mathbf{A}\}, \mathtt{T}\rangle\rangle, \langle \neg\mathbf{A}, \langle \{\neg\mathbf{A}\}, \mathtt{F}\rangle\rangle \in \mathcal{D}_o$. Let $\mathsf{p} := \langle \mathbf{A}, \langle \{\mathbf{A}\}, \mathtt{T}\rangle\rangle$ and $\mathsf{q} := \langle \neg\mathbf{A}, \langle \{\neg\mathbf{A}\}, \mathtt{F}\rangle\rangle$. Now, in either case, $v(\mathsf{p}) \equiv \mathtt{T}$ and $v(\mathsf{q}) \equiv \mathtt{F}$, and v is surjective.

So, $\mathcal{P} := (\mathcal{A}, \sim, v)$ is a Σ-per evaluation. We now check $\mathcal{P}$ has each of the desired properties.

$\partial^{o,\iota}$: On base types, $\sim$ is a total equivalence relation by definition.

∂^Σ: For each constant $c \in \Sigma$, $\mathcal{E}(c) \equiv \mathcal{I}(c) \sim \mathcal{I}(c) \equiv \mathcal{E}(c)$.

∂^c: By the definition of $\sim$ on function domains $\mathcal{D}_{\alpha\to\beta}$, for each $\mathsf{g},\mathsf{h} \in \mathcal{D}_{\alpha\to\beta}$ and $\mathsf{a},\mathsf{b} \in \mathcal{D}_\alpha$, if $\mathsf{g} \sim \mathsf{h}$ and $\mathsf{a} \sim \mathsf{b}$, then $\mathsf{g}@\mathsf{a} \sim \mathsf{h}@\mathsf{b}$.

∂^s: We have prove $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\psi(\mathbf{A})$ whenever $\varphi \sim \psi$ for every $\mathbf{A} \in \mathit{wff}_\alpha$ by induction on $\mathbf{A}$. This follows from $\varphi \sim \psi$ if $\mathbf{A}$ is a variable. If $\mathbf{A}$ is a constant c, this follows from the assumption that $\mathcal{E}_\varphi(c) \in \overline{\mathcal{D}^c_\alpha}$. If $\mathbf{A}$ is an application $(\mathbf{GB})$, then by induction we know $\mathcal{E}_\varphi(\mathbf{G}) \sim \mathcal{E}_\psi(\mathbf{G})$ and $\mathcal{E}_\varphi(\mathbf{B}) \sim \mathcal{E}_\psi(\mathbf{B})$. By the definition of $\sim$ at function types, we have $\mathcal{E}_\varphi(\mathbf{GB}) \sim \mathcal{E}_\psi(\mathbf{GB})$.

Suppose $\mathbf{A}$ is a λ-abstraction $\lambda X_\beta \mathbf{.} \mathbf{C}_\gamma$. By induction, we know

$$\mathcal{E}_\varphi(\mathbf{A})@\mathsf{a} \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{C}) \sim \mathcal{E}_{\psi,[\mathsf{b}/X]}(\mathbf{C}) \equiv \mathcal{E}_\psi(\mathbf{A})@\mathsf{b}$$

whenever $\mathsf{a},\mathsf{b} \in \mathcal{D}_\beta$ and $\mathsf{a} \sim \mathsf{b}$. In order to show $\mathcal{E}_\varphi(\mathbf{A}) \sim \mathcal{E}_\psi(\mathbf{A})$, we must show $\neg(\mathbf{A}' \doteq \mathbf{A}'') \notin \mathcal{H}_{\xi*}$ where $\mathbf{A}'$ is the first component of $\mathcal{E}_\varphi(\mathbf{A})$ and $\mathbf{A}''$ is the first component of $\mathcal{E}_\psi(\mathbf{A})$. Let θ^φ and θ^ψ be the substitutions defined on $\mathit{free}(\mathbf{A})$ where $\theta^\varphi(x)$ and $\theta^\psi(x)$ are the first components of $\varphi(x)$ and $\psi(x)$, resp. By Theorem 2.4.17, we know $\mathbf{A}'$ is $\theta^\varphi(\mathbf{A}){\downarrow_\beta}$ and $\mathbf{A}''$ is $\theta^\psi(\mathbf{A}){\downarrow_\beta}$. So, $\mathbf{A}'$ is $\lambda X_\beta \mathbf{.} \theta^\varphi(\mathbf{C}){\downarrow_\beta}$ and $\mathbf{A}''$ is $\lambda X_\beta \mathbf{.} \theta^\psi(\mathbf{C}){\downarrow_\beta}$. Assume $\neg(\mathbf{A}' \doteq \mathbf{A}'') \in \mathcal{H}_{\xi*}$. By $\vec{\nabla}_\xi$, there must be some parameter w such that $\neg((\theta^\varphi,[w/X])(\mathbf{C}){\downarrow_\beta} \doteq (\theta^\psi,[w/X](\mathbf{C})){\downarrow_\beta}) \in \mathcal{H}_{\xi*}$. This implies $(\theta^\varphi,[w/X])(\mathbf{C})$ and $(\theta^\psi,[w/X])(\mathbf{C})$ are ξ-incompatible. Let $\varphi' := \varphi,[\mathcal{E}(w)/X]$ and $\psi' := \psi,[\mathcal{E}(w)/X]$. So, $\theta^{\varphi'}(\mathbf{C})$ and $\theta^{\psi'}(\mathbf{C})$ are ξ-incompatible. But this contradicts $\mathcal{E}_{\varphi'}(\mathbf{C}) \sim \mathcal{E}_{\psi'}(\mathbf{C})$ and Lemma 2.4.43:1.

∂^v: We have $\mathsf{a} \sim \mathsf{b}$ implies $\upsilon(\mathsf{a}) \equiv \upsilon(\mathsf{b})$ for each $\mathsf{a},\mathsf{b} \in \mathcal{D}_o$ by the definition of $\sim$ at type o.

$\partial^\neg$: This follows from the definition of $n : \mathcal{D}_o \to \mathcal{D}_o$ above.

$\partial^\vee$: This follows from the definition of d above.

∂^Π: This follows from the definition of π^α (which was relativized to the smaller domains $\overline{\mathcal{D}_\alpha}$) above.

∂^ξ: Let $\mathbf{M},\mathbf{N} \in \mathit{wff}_\beta(\Sigma)$, an assignments φ with $\varphi \sim \varphi$, and a variable X_α be given. Suppose $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \sim \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N})$ for every $\mathsf{a} \in \overline{\mathcal{D}^\sim_\alpha}$. For any $\mathsf{a},\mathsf{b} \in \overline{\mathcal{D}^\sim_\alpha}$ with $\mathsf{a} \sim \mathsf{b}$, we can use this with ∂^s (shown above) to conclude

$$\mathcal{E}_\varphi(\lambda X\mathbf{.M})@\mathsf{a} \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{M}) \sim \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{N}) \sim \mathcal{E}_{\varphi,[\mathsf{b}/X]}(\mathbf{N}) \equiv \mathcal{E}_\varphi(\lambda X\mathbf{.N})@\mathsf{b}.$$

Assume $\mathcal{E}_\varphi(\lambda X\mathbf{.M}) \not\sim \mathcal{E}_\varphi(\lambda X\mathbf{.N})$. The only way this is possible is if $\neg(\theta^\varphi(\lambda X\mathbf{.M}){\downarrow_\beta} \doteq \theta^\varphi(\lambda X\mathbf{.N}){\downarrow_\beta}) \in \mathcal{H}_{\xi*}$. But in this case, we can use ξ to conclude $\neg(\theta^\varphi([w/X]\mathbf{M}){\downarrow_\beta} \doteq \theta^\varphi([w/X]\mathbf{N}){\downarrow_\beta}) \in \mathcal{H}_{\xi*}$ for some parameter w_α. This contradicts

$$\mathcal{E}_\varphi(\lambda X\mathbf{.M})@\mathcal{E}(w) \sim \mathcal{E}_\varphi(\lambda X\mathbf{.N})@\mathcal{E}(w)$$

and Lemma 2.4.43:1.

$\partial^{\mathfrak{q}}$: We will use the fact that $\mathcal{H}_{\xi*}$ is equation-free. Let a type α be given. We need to find an element $\mathrm{q}^\alpha \in \mathcal{D}_{\alpha\to\alpha\to o}$ such that for all $\mathsf{a},\mathsf{b} \in \overline{\mathcal{D}^\sim_\alpha}$, $\upsilon(\mathrm{q}^\alpha@\mathsf{a}@\mathsf{b}) \equiv \mathtt{T}$ iff $\mathsf{a} \sim \mathsf{b}$. We distinguish between the $\mathcal{H}_{\beta\xi}$ and $\mathcal{H}_{\beta\xi\mathfrak{b}}$ cases.

$\mathcal{H}_{\beta\xi\mathfrak{b}}$**:** For each $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$, let $s_\mathsf{a} : \mathcal{D}_\alpha \to \mathcal{D}_o$ be defined by $s_\mathsf{a}(\mathsf{b}) := \langle (\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \mathtt{T}\rangle$ for $\mathsf{b} \equiv \langle \mathbf{B}, b\rangle$ if $\mathsf{b} \sim \mathsf{a}$ in $\mathcal{D}_\alpha$, and $s_\mathsf{a}(\mathsf{b}) := \langle (\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \mathtt{F}\rangle$ otherwise. To show this is well-defined, we need to know $s_\mathsf{a}(\mathsf{b})$ really is in $\mathcal{D}_o$. If $\mathsf{a} \sim \mathsf{b}$ and $s_\mathsf{a}(\mathsf{b}) \equiv ((\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \mathtt{T}) \notin \mathcal{D}_o$, then we must have $\neg(\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta} \in \mathcal{H}_{\beta\xi\mathfrak{b}}$. This contradicts $\mathsf{a} \sim \mathsf{b}$ by Lemma 2.4.43:1. If $\mathsf{a} \not\sim \mathsf{b}$ and $s_\mathsf{a}(\mathsf{b}) \equiv ((\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \mathtt{F}) \notin \mathcal{D}_o$, then we must have $(\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta} \in \mathcal{H}_{\beta\xi\mathfrak{b}}$, contradicting the assumption that $\mathcal{H}_{\beta\xi\mathfrak{b}}$ is equation-free.[27] Note that this includes the case where $\mathsf{a} \not\sim \mathsf{a}$. So we now have $\langle (\lambda X_\alpha . \mathbf{A} \doteq^\alpha x){\downarrow_\beta}, s_\mathsf{a}\rangle \in \mathcal{D}_{\alpha\to o}$. EdNote(27)

Now, define $l : \mathcal{D}_\alpha \to \mathcal{D}_{\alpha\to o}$ by $l(\mathsf{a}) := \langle (\lambda X_\alpha . \mathbf{A} \doteq^\alpha x){\downarrow_\beta}, s_\mathsf{a}\rangle$ for any $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$. Since each $s_\mathsf{a} \in \mathcal{D}_{\alpha\to o}$, this is well-defined. Let $\mathrm{q}^\alpha := \langle \doteq^\alpha, l\rangle$. We must check $\mathrm{q}^\alpha \sim \mathrm{q}^\alpha$. Let $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$ and $\mathsf{b} \equiv \langle \mathbf{B}, b\rangle \in \mathcal{D}_\alpha$ with $\mathsf{a} \sim \mathsf{b}$ be given. We need to check $l(\mathsf{a}) \sim l(\mathsf{b})$. That is, we must check that s_a and s_b sends $\sim$-related values to $\sim$-related results. Let $\mathsf{c} \equiv \langle \mathbf{C}, c\rangle \in \mathcal{D}_\alpha$ and $\mathsf{d} \equiv \langle \mathbf{D}, d\rangle \in \mathcal{D}_\alpha$ with $\mathsf{c} \sim \mathsf{d}$ be given. $\upsilon(s_\mathsf{a}(\mathsf{c})) \equiv \mathtt{T}$ iff $\mathsf{a} \sim \mathsf{c}$ iff $\mathsf{b} \sim \mathsf{d}$ iff $\upsilon(s_\mathsf{b}(\mathsf{d})) \equiv \mathtt{T}$. So, $s_\mathsf{a}(\mathsf{c}) \sim s_\mathsf{b}(\mathsf{d})$.

$\mathcal{H}_{\beta\xi}$**:** Since $\mathcal{D}_o$ was defined differently in this case, we must define each s_a more carefully. For each $\mathsf{a} \in \mathcal{D}_\alpha$, define $S_\mathsf{a} := \{\mathbf{C} | \exists c . \langle \mathbf{C}, c\rangle \sim \mathsf{a}\}$. This is compatible by Lemma 2.4.43:1. For any $\mathsf{a}, \mathsf{b} \in \mathcal{D}_\alpha$, $S_\mathsf{a} \doteq^\alpha S_\mathsf{b}$ is compatible by Lemma 2.4.33.

Let $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$ be given. For each $\mathsf{b} \equiv \langle \mathbf{B}, b\rangle \in \mathcal{D}_\alpha$, define $s_\mathsf{a}(\mathsf{b}) := \langle (\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \langle S_\mathsf{a} \doteq^\alpha S_\mathsf{b}, \mathtt{T}\rangle\rangle$ if $\mathsf{a} \sim \mathsf{b}$, and $s_\mathsf{a}(\mathsf{b}) := \langle (\mathbf{A} \doteq^\alpha \mathbf{B}){\downarrow_\beta}, \langle S_\mathsf{a} \doteq^\alpha S_\mathsf{b}, \mathtt{F}\rangle\rangle$ otherwise. The verification that s_a is well-defined is just as in the $\mathcal{H}_{\beta\xi\mathfrak{b}}$ case. As before, we define $\mathrm{q}^\alpha := \langle \doteq^\alpha, l\rangle$ where $l : \mathcal{D}_\alpha \to \mathcal{D}_{\alpha\to o}$ is defined by $l(\mathsf{a}) := \langle (\lambda X_\alpha . \mathbf{A} \doteq^\alpha x){\downarrow_\beta}, s_\mathsf{a}\rangle$ for any $\mathsf{a} \equiv \langle \mathbf{A}, a\rangle \in \mathcal{D}_\alpha$. The verification that $\mathrm{q}^\alpha \sim \mathrm{q}^\alpha$ again reduces to showing that if $\mathsf{a} \sim \mathsf{b}$ and $\mathsf{c} \sim \mathsf{d}$, then $s_\mathsf{a}(\mathsf{c}) \sim s_\mathsf{b}(\mathsf{d})$, but this time we have more to check to verify $\sim$ at type o.

Let $\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d} \in \mathcal{D}_\alpha$ with $\langle \mathbf{A}, a\rangle \equiv \mathsf{a} \sim \mathsf{b} \equiv \langle \mathbf{B}, b\rangle$, $\langle \mathbf{C}, c\rangle \equiv \mathsf{c} \sim \mathsf{d} \equiv \langle \mathbf{D}, d\rangle$ be given. By definition, we have $s_\mathsf{a}(\mathsf{c}) \equiv \langle (\mathbf{A} \doteq^\alpha \mathbf{C}){\downarrow_\beta}, \langle S_\mathsf{a} \doteq^\alpha S_\mathsf{c}, p\rangle\rangle$ and $s_\mathsf{b}(\mathsf{d}) \equiv \langle (\mathbf{B} \doteq^\alpha \mathbf{D}){\downarrow_\beta}, \langle S_\mathsf{b} \doteq^\alpha S_\mathsf{d}, q\rangle\rangle$. We can easily show $p \equiv q$ as before, since $p \equiv \mathtt{T}$ iff $\mathsf{a} \sim \mathsf{c}$ iff $\mathsf{b} \sim \mathsf{d}$ iff $q \equiv \mathtt{T}$. It remains to show $(S_\mathsf{a} \doteq^\alpha S_\mathsf{c}) \equiv (S_\mathsf{b} \doteq^\alpha S_\mathsf{d})$. This follows from $S_\mathsf{a} \equiv S_\mathsf{b}$ (since $\mathsf{a} \sim \mathsf{b}$) and $S_\mathsf{c} \equiv S_\mathsf{d}$ (since $\mathsf{c} \sim \mathsf{d}$).

This verifies $\partial^\mathfrak{q}$ in both cases.

Furthermore, in the $\mathcal{H}_{\beta\xi\mathfrak{b}}$ case, we clearly have only two $\sim$-equivalence classes in $\mathcal{D}_o$ since there are only two possibilities for the second component of each $\mathsf{a} \in \mathcal{D}_o$. So, we have $\partial^\mathfrak{b}$ in this case.

We can now apply Theorem 2.4.10 to obtain a model $\mathcal{M} \equiv (\mathcal{D}^\sim, @^\sim, \mathcal{E}^\sim, \upsilon^\sim)$ in $\mathfrak{M}_{\beta\xi\mathfrak{b}}$ $[\mathfrak{M}_{\beta\xi}]$ such that $\upsilon^\sim(\mathcal{E}^\sim(\mathbf{A})) \equiv \upsilon(\mathcal{E}(\mathbf{A}))$ for all $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$. Note that $\mathcal{M} \models \mathcal{H}_{\xi*}$ since for each $\mathbf{A} \in \mathcal{H}_{\xi*}$ $\upsilon^\sim(\mathcal{E}^\sim(\mathbf{A})) \equiv \upsilon(\mathcal{E}(\mathbf{A})) \equiv \mathtt{T}$. □

2.4.10 Model Existence for $\mathfrak{Acc}_{\beta\mathfrak{b}}$ and $\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$

The cases where we require boolean, but not functional, extensionality must be handled in a different way. Attempts to use a per evaluation fail. On the other hand, we cannot directly use a possible values structure since $\mathcal{D}_o$ cannot have only two values in a possible values structure. Instead, we use a modification of the Andrews structure with a compatibility relation appropriate for this case.

[27] EdNote: this is where we need $\mathcal{H}_{\beta\xi\mathfrak{b}}$ is equation-free

We can argue the $\mathfrak{Acc}_{\beta\mathfrak{b}}$ and $\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$ cases simultaneously, using the term "normal" and the notation $\mathbf{A}{\downarrow_*}$ as before.

Definition 2.4.46 ($\mathcal{H}$-Boolean Compatibility) Let $\mathcal{H}$ be a Hintikka set in $\mathfrak{Acc}_{\beta}$. We say two closed terms $\mathbf{A}, \mathbf{B} \in cwff_\alpha(\Sigma)$ are *$\mathcal{H}$-Boolean compatible* (written $\mathbf{A} \mathrel{\|^{\flat}} \mathbf{B}$) if there is a term $\mathbf{C} \in wff_\alpha(\Sigma)$ with $free(\mathbf{C}) \equiv \{y_o^1, \ldots, y_o^n\}$ and normal closed terms $\mathbf{P}_o^1, \mathbf{Q}_o^1, \ldots, \mathbf{P}_o^n, \mathbf{Q}_o^n$ such that

1. $\mathbf{A}{\downarrow_*} \equiv \theta(\mathbf{C}){\downarrow_*}$,
2. $\mathbf{B}{\downarrow_*} \equiv \psi(\mathbf{C}){\downarrow_*}$ and
3. for each $1 \leq i \leq n$, $\{\mathbf{P}^i, \neg\mathbf{Q}^i\} \not\subseteq \mathcal{H}$ and $\{\neg\mathbf{P}^i, \mathbf{Q}^i\} \not\subseteq \mathcal{H}$

where $\theta(y^i) \equiv \mathbf{P}^i$ and $\psi(y^i) \equiv \mathbf{Q}^i$ for each $1 \leq i \leq n$. As before, we say a set S of terms is $\mathcal{H}$-Boolean compatible if $\mathbf{A} \mathrel{\|^{\flat}} \mathbf{B}$ for every pair $\mathbf{A}, \mathbf{B} \in S$.

Lemma 2.4.47 *For each closed $\mathbf{A}, \mathbf{B} \in cwff_\alpha(\Sigma)$, we have $\mathbf{A} \mathrel{\|^{\flat}} \mathbf{B}$ iff $\mathbf{A}{\downarrow_*} \mathrel{\|^{\flat}} \mathbf{B}{\downarrow_*}$.*

Proof: This is immediate since we defined $\|^{\flat}$ in terms of normal forms. $\Box$

Lemma 2.4.48 *For each closed $\mathbf{A} \in cwff_\alpha(\Sigma)$, we have $\mathbf{A} \mathrel{\|^{\flat}} \mathbf{A}$.*

Proof: This is immediate, since $\mathbf{A}$ can act as the witness (with $n = 0$) that $\mathbf{A} \mathrel{\|^{\flat}} \mathbf{A}$. $\Box$

Lemma 2.4.49 *If $\mathbf{F}_{\alpha\to\beta} \mathrel{\|^{\flat}} \mathbf{G}_{\alpha\to\beta}$ and $\mathbf{A}_\alpha \mathrel{\|^{\flat}} \mathbf{B}_\alpha$, then $(\mathbf{FA}) \mathrel{\|^{\flat}} (\mathbf{GB})$.*

Proof: We must have terms $\mathbf{H} \in wff_\alpha(\Sigma)$ and $\mathbf{C} \in wff_\alpha(\Sigma)$ and closed terms $\mathbf{P}_o^1, \mathbf{Q}_o^1, \ldots, \mathbf{P}_o^n, \mathbf{Q}_o^n$ and $\mathbf{P}_o^{n+1}, \mathbf{Q}_o^{n+1}, \ldots, \mathbf{P}_o^{n+m}, \mathbf{Q}_o^{n+m}$ as in the definition of $\|^{\flat}$. Without loss of generality, we can assume $free(\mathbf{H}) \cap free(\mathbf{C}) \equiv \emptyset$ (otherwise, we can rename the variables in $\mathbf{C}$). Then, $(\mathbf{HC})$ and $\mathbf{P}_o^1, \mathbf{Q}_o^1, \ldots, \mathbf{P}_o^{n+m}, \mathbf{Q}_o^{n+m}$ witness that $(\mathbf{FA}) \mathrel{\|^{\flat}} (\mathbf{GB})$. $\Box$

Lemma 2.4.50 *Let Γ_Σ be an acceptable abstract consistency class in $\mathfrak{Acc}_{\beta\mathfrak{b}}$ or $\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$. Let $\mathcal{H}$ be a Hintikka set in Γ_Σ. If $\mathbf{C} \in wff_\alpha(\Sigma)$ is a normal term with $free(\mathbf{C}) \equiv \{y_o^1, \ldots, y_o^n\}$ and θ and ψ are substitutions with $\theta(y^i), \psi(y^i) \in cwff_o(\Sigma)$ normal terms such that $\{\theta(y^i), \neg\psi(y^i)\} \not\subseteq \mathcal{H}$ and $\{\neg\theta(y^i), \psi(y^i)\} \not\subseteq \mathcal{H}$ for each $1 \leq i \leq n$. Then, $\neg(\theta(\mathbf{C}){\downarrow_*} \doteq \psi(\mathbf{C}){\downarrow_*}) \notin \mathcal{H}$. Furthermore, if $\alpha \equiv o$, then $\{\theta(\mathbf{C}){\downarrow_*}, \neg\psi(\mathbf{C}){\downarrow_*}\} \not\subseteq \mathcal{H}$ and $\{\neg\theta(\mathbf{C}){\downarrow_*}, \psi(\mathbf{C}){\downarrow_*}\} \not\subseteq \mathcal{H}$.*

Proof: We prove this by induction on the "depth" of the normal term $\mathbf{C}$. Note that $\mathbf{C}$ must have the form $\lambda\overline{X^m}.h\overline{\mathbf{A}^k}$ where h is a variable or $h \in \Sigma \cup \overline{x^m}$. We define the depth inductively by

$$depth(\lambda\overline{X^m}.h\overline{\mathbf{A}^k}) := 1 + \max_{i=1,\ldots,k} depth(\lambda\overline{X^m}.\mathbf{A}^i).$$

Note that if $k = 0$, then $depth(\lambda\overline{X^m}.h) \equiv 1$. It is easy to show that if $\mathbf{A}_{\beta\to o}$ is a normal term and $w_\beta \in \Sigma$, then $depth(\mathbf{A}) \equiv depth((\mathbf{A}w){\downarrow_*})$. So, $depth(\Pi\mathbf{A}) \equiv 1 + depth(\mathbf{A}w)$. This will be important when we consider when h is Π.

Since Γ_Σ is acceptable, $\mathcal{H}$ satisfies $\vec{\nabla}_m$, $\vec{\nabla}_{\mathfrak{b}}$, $\vec{\nabla}_{fdec}$, $\vec{\nabla}_k$ and $\vec{\nabla}_{dec}^w$ for every parameter w, as well as the properties Hintikka sets satisfy in every abstract consistency class.

If h is a variable, it must be y_o^i for some i (and k must be 0). Assume we have $\neg(\lambda\overline{X^m}.\theta(y^i) \doteq \lambda\overline{X^m}.\psi(y^i)) \in \mathcal{H}$. We can apply $\vec{\nabla}_k$ m times to obtain $\neg(\theta(y^i) \doteq \theta(y^i)) \in \mathcal{H}$. Then by $\vec{\nabla}_{\mathfrak{b}}$, we have $\{\theta(y^i), \neg\psi(y^i)\} \subseteq \mathcal{H}$ or $\{\neg\theta(y^i), \psi(y^i)\} \subseteq \mathcal{H}$ contradicting our assumption about θ and ψ

Suppose $h \in \Sigma \cup \overline{x^m}$ and assume $\neg(\theta(\lambda\overline{X^m}.h\overline{\mathbf{A}^k}){\downarrow_*} \doteq \psi(\lambda\overline{X^m}.h\overline{\mathbf{A}^k}){\downarrow_*}) \in \mathcal{H}$. Since $\vec{\nabla}_{fdec}$ only applies at function types, we consider two cases.

Suppose $\mathbf{C}$ is of base type, so $m = 0$ and $h \in \Sigma$. If h is a parameter, then we can apply $\vec{\nabla}^h_{dec}$ to obtain $\neg(\theta(\mathbf{A}^i){\downarrow_*} \doteq \psi(\mathbf{A}^i){\downarrow_*}) \in \mathcal{H}$ for some $1 \leq i \leq n$. This contradicts the induction hypothesis with $\mathbf{A}^i$. Furthermore, if $h\overline{A^k}$ is of type o, then $\vec{\nabla}_m$ implies $\{\theta(\mathbf{C}){\downarrow_*}, \neg\psi(\mathbf{C}){\downarrow_*}\} \not\subseteq \mathcal{H}$ and $\{\neg\theta(\mathbf{C}){\downarrow_*}, \psi(\mathbf{C}){\downarrow_*}\} \not\subseteq \mathcal{H}$.

If h is $\neg$, then $\vec{\nabla}_\mathfrak{b}$ and $\overline{\nabla}_\neg$ imply without loss of generality $\theta(\mathbf{A}^1){\downarrow_*}, \neg\psi(\mathbf{A}^1){\downarrow_*} \in \mathcal{H}$, contradicting the inductive hypothesis for $\mathbf{A}^1$. Similarly, if h is $\vee$, then $\vec{\nabla}_\mathfrak{b}$, $\overline{\nabla}_\vee$ and $\overline{\nabla}_\wedge$ contradict the inductive hypothesis. If h is Π^β, then $\vec{\nabla}_\mathfrak{b}$, $\overline{\nabla}_\exists$ and $\overline{\nabla}_\forall$ imply (without loss of generality) there is a parameter w_β with $\theta(\mathbf{A}^1 w){\downarrow_*}, \neg\psi(\mathbf{A}^1 w){\downarrow_*} \in \mathcal{H}$. This contradicts the inductive hypothesis for $(\mathbf{A}^1 w){\downarrow_*}$ since $depth((\mathbf{A}^1 w){\downarrow_*}) < depth(\Pi^\beta \mathbf{A}^1)$.

Now, suppose $\mathbf{C}$ is of function type. We can apply $\vec{\nabla}_{fdec}$ to obtain $\neg(\theta(\lambda\overline{X^m}.\mathbf{A}^i){\downarrow_*} \doteq \psi(\lambda\overline{X^m}.\mathbf{A}^i){\downarrow_*}) \in \mathcal{H}$. for some $1 \leq i \leq n$. This contradicts the induction hypothesis with the smaller normal term $\lambda\overline{X^m}.\mathbf{A}^i$. ☐

Lemma 2.4.51 *Let Γ_Σ be an acceptable abstract consistency class in $\mathfrak{Acc}_{\beta\mathfrak{b}}$ or $\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$. Let $\mathcal{H}$ be a Hintikka set in Γ_Σ. If $\mathbf{A} \|^{\flat} \mathbf{B}$ for $\mathbf{A}, \mathbf{B} \in cwff_\alpha(\Sigma)$, then $\neg(\mathbf{A} \doteq \mathbf{B}){\downarrow_*} \notin \mathcal{H}$. Furthermore, if $\alpha \equiv o$, then $\{\mathbf{A}, \neg\mathbf{B}\} \not\subseteq \mathcal{H}$ and $\{\neg\mathbf{A}, \mathbf{B}\} \not\subseteq \mathcal{H}$.*

Proof: This follows by applying Lemma 2.4.50 to the witnesses $\mathbf{C}$, θ and ψ of $\mathbf{A} \|^{\flat} \mathbf{B}$. ☐

We now define the structures of our models.

- o Let $\mathcal{D}_o := \{\langle\mathcal{H}^{\mathtt{T}}, \mathtt{T}\rangle, \langle\mathcal{H}^{\mathtt{F}}, \mathtt{F}\rangle\}$ where $\mathcal{H}^{\mathtt{T}} := \{\mathbf{A} \in cwff_o(\Sigma){\downarrow_*} \mid \neg\mathbf{A} \notin \mathcal{H}\}$ and $\mathcal{H}^{\mathtt{F}} := \{\mathbf{A} \in cwff_o(\Sigma){\downarrow_*} \mid \mathbf{A} \notin \mathcal{H}\}$.
- ι Let $\mathcal{D}_\iota := \{\langle S, *\rangle \mid \emptyset \not\equiv S \subseteq cwff_\iota(\Sigma){\downarrow_*}$ is $\mathcal{H}$-Boolean Compatible$\}$.
- $\beta \to \gamma$ Let $\mathcal{D}_{\beta\to\gamma} := \{\langle F, f\rangle \mid \emptyset \not\equiv F \subseteq cwff_{\beta\to\gamma}(\Sigma){\downarrow_*}$ $\mathcal{H}$-Boolean Compatible$\}$ where $f : \mathcal{D}_\beta \to \mathcal{D}_\gamma$ is such that if $\langle F, f\rangle \in \mathcal{D}_{\beta\to\gamma}$ and $f(\langle B, b\rangle) \equiv \langle C, c\rangle$, then $(\mathbf{FB}){\downarrow_*} \in C$ for every $\mathbf{F} \in F$ and $\mathbf{B} \in B$.

We define @ by $\langle F, f\rangle @ \langle B, b\rangle := f(\langle B, b\rangle)$.

Lemma 2.4.52 *For every $\mathcal{H}$-Boolean Compatible set $S \subseteq cwff_\alpha(\Sigma){\downarrow_*}$, there is a set $S' \supseteq S$ and value s such that $\langle S', s\rangle \in \mathcal{D}_\alpha$.*

Proof: We prove this by induction on types.

- o We must have $S \subseteq \mathcal{H}^{\mathtt{T}}$ or $S \subseteq \mathcal{H}^{\mathtt{F}}$. If not, then there are $\mathbf{A}, \mathbf{B} \in S$ such that $\mathbf{A} \|^{\flat} \mathbf{B}$, $\mathbf{A} \in \mathcal{H}$ and $\neg\mathbf{A} \in \mathcal{H}$. This contradicts Lemma 2.4.51.
- ι Here we immediately have $\langle S, *\rangle \in \mathcal{D}_o$.
- $\beta \to \gamma$ Here we simply define an appropriate function $f : \mathcal{D}_\beta \to \mathcal{D}_\gamma$ such that $\langle S, f\rangle \in \mathcal{D}_{\alpha\to\gamma}$. For each $\langle B, b\rangle \in \mathcal{D}_\beta$, let $F_{\langle B,b\rangle} := \{(\mathbf{FB}){\downarrow_*} \mid \mathbf{F} \in S, \mathbf{B} \in B\}$. This set is $\mathcal{H}$-Boolean Compatible by Lemma 2.4.49. So, by the induction hypothesis at γ, there must be some $F'_{\langle B,b\rangle} \supset F_{\langle B,b\rangle}$ and value $f_{\langle B,b\rangle}$ with $\langle F'_{\langle B,b\rangle}, f_{\langle B,b\rangle}\rangle \in \mathcal{D}_\gamma$. Using the axiom of choice, we define $f(\langle B, b\rangle) := \langle F'_{\langle B,b\rangle}, f_{\langle B,b\rangle}\rangle$. It is easy to verify $\langle S, f\rangle \in \mathcal{D}_{\beta\to\gamma}$. ☐

Lemma 2.4.53 *If θ and ψ are substitutions such that $\theta(x) \|^{\mathfrak{b}} \psi(x)$ for each variable x, then for every $\mathbf{A} \in \mathit{wff}_\alpha(\Sigma)$, $\theta(\mathbf{A}) \|^{\mathfrak{b}} \psi(\mathbf{A})$.*

Proof: Let $\{x^1, \ldots, x^m\}$ be the set of free variables in $\mathbf{A}$. For each x^j, there is a $\mathbf{C}^j$ with $free(\mathbf{C}^j) \equiv \{y_o^{j,1}, \ldots, y_o^{j,n_j}\}$ and normal closed terms $\mathbf{P}_o^{j,1}, \mathbf{Q}_o^{j,1}, \ldots, \mathbf{P}_o^{j,n_j}, \mathbf{Q}_o^{j,n_j}$ which witness $\theta(x^j) \|^{\mathfrak{b}} \psi(x^j)$. Without loss of generality, we may assume each $y^{j,i}$ is distinct and not a member of $free(\mathbf{A})$. Let φ be a substitution with $\varphi(x^j) := \mathbf{C}^j$ and let $\mathbf{C} := \varphi(\mathbf{A})$. We can now show that $\mathbf{C}$, $\{y^{j,i} | 1 \leq j \leq m, 1 \leq i \leq n_j\}$ and $\{\mathbf{P}_o^{j,i}, \mathbf{Q}_o^{j,i} | 1 \leq j \leq m, 1 \leq i \leq n_j\}$ witness $\theta(\mathbf{A}) \|^{\mathfrak{b}} \psi(\mathbf{A})$. We have $\{\mathbf{P}^{j,i}, \neg\mathbf{Q}^{j,i}\} \not\subseteq \mathcal{H}$ and $\{\neg\mathbf{P}^{j,i}, \mathbf{Q}^{j,i}\} \not\subseteq \mathcal{H}$ for each j and i since each $\mathbf{P}^{j,i}$ and $\mathbf{Q}^{j,i}$ witnessed $\theta(x^j) \|^{\mathfrak{b}} \psi(x^j)$. Let θ' and ψ' be substitutions such that $\theta'(y^{j,i}) := \mathbf{P}^{j,i}$ and $\psi'(y^{i,j}) := \mathbf{Q}^{j,i}$. We have $\theta'(\mathbf{C}){\downarrow_*} \equiv \theta'(\varphi(\mathbf{A})){\downarrow_*} \equiv \theta(\mathbf{A}){\downarrow_*}$ and $\psi'(\mathbf{C}){\downarrow_*} \equiv \psi'(\varphi(\mathbf{A})){\downarrow_*} \equiv \psi(\mathbf{A}){\downarrow_*}$. So, $\theta(\mathbf{A}) \|^{\mathfrak{b}} \psi(\mathbf{A})$. $\square$

Theorem 2.4.54 (Model Existence ($\mathfrak{Acc}_{\beta\mathfrak{b}}$, $\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$)) *Let $\mathcal{H}$ be an acceptable Hintikka set in $\mathfrak{Hint}_{\beta\mathfrak{b}}$ [$\mathfrak{Hint}_{\beta\eta\mathfrak{b}}$]. There is a model $\mathcal{M}$ in $\mathfrak{M}_{\beta\mathfrak{b}}$ [$\mathfrak{M}_{\beta\eta\mathfrak{b}}$] such that $\mathcal{M} \models \mathcal{H}$.*

Proof: We start with the structure $(\mathcal{D}, @)$ already defined. We can apply Lemma 2.4.52 with $\{c\}$ to define $\mathcal{E}(c)$ for parameters $c_\alpha \in \Sigma$ to be any $\langle S, s\rangle$ in $\mathcal{D}_\alpha$ such that $c \in S$. For logical constants, we need to show the desired interpretations are in the structure.

Let $n : o \to o$ be defined by $n(\langle \mathcal{H}^{\mathtt{T}}, \mathtt{T}\rangle) := (\langle \mathcal{H}^{\mathtt{F}}, \mathtt{F}\rangle)$ and $n(\langle \mathcal{H}^{\mathtt{F}}, \mathtt{F}\rangle) := (\langle \mathcal{H}^{\mathtt{T}}, \mathtt{T}\rangle)$. To check $\langle \{\neg\}, n\rangle \in \mathcal{D}_{o\to o}$, let $\mathbf{A} \in \mathcal{H}^{\mathtt{T}}$ be given. So, $\neg\mathbf{A} \notin \mathcal{H}$ and $\neg\mathbf{A} \in \mathcal{H}^{\mathtt{F}}$. On the other hand, if $\mathbf{A} \in \mathcal{H}^{\mathtt{F}}$,then $\mathbf{A} \notin \mathcal{H}$. So, by $\overline{\nabla}_\neg$, $\neg\neg\mathbf{A} \notin \mathcal{H}$ and $\neg\mathbf{A} \in \mathcal{H}^{\mathtt{T}}$. Thus, we can let $\mathcal{E}(\neg) := \langle \{\neg\}, n\rangle$.

For disjunction, let $d_{\mathtt{T}} : \mathcal{D}_o \to \mathcal{D}_o$ be defined by $d_{\mathtt{T}}(\langle \mathcal{H}^{\mathsf{p}}, \mathsf{p}\rangle) := (\langle \mathcal{H}^{\mathtt{T}}, \mathtt{T}\rangle)$ for $\mathsf{p} \in \{\mathtt{T}, \mathtt{F}\}$. To check that $\langle \{\vee\mathbf{A} | \mathbf{A} \in \mathcal{H}^{\mathtt{T}}\}, d_{\mathtt{T}}\rangle \in \mathcal{D}_{o\to o}$, note that for any $\mathbf{A} \in \mathcal{H}^{\mathtt{T}}$ and $\mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\neg\mathbf{A} \notin \mathcal{H}$ and $\overline{\nabla}_\wedge$ implies $\neg(\mathbf{A} \vee \mathbf{B}) \notin \mathcal{H}$. So, $\mathbf{A} \vee \mathbf{B} \in \mathcal{H}^{\mathtt{T}}$. Next, let $d_{\mathtt{F}} : \mathcal{D}_o \to \mathcal{D}_o$ be the identity function. To check $\langle \{\vee\mathbf{A} | \mathbf{A} \in \mathcal{H}^{\mathtt{F}}\}, d_{\mathtt{F}}\rangle \in \mathcal{D}_{o\to o}$, note that if $\mathbf{A} \in \mathcal{H}^{\mathtt{F}}$ and $\mathbf{B} \in \mathcal{H}^{\mathtt{T}}$, then $\neg\mathbf{B} \notin \mathcal{H}$ and $\overline{\nabla}_\wedge$ implies $\neg(\mathbf{A} \vee \mathbf{B}) \notin \mathcal{H}$ and $\mathbf{A} \vee \mathbf{B} \in \mathcal{H}^{\mathtt{T}}$. Also, if $\mathbf{A}, \mathbf{B} \in \mathcal{H}^{\mathtt{F}}$, then $\mathbf{A} \notin \mathcal{H}$ and $\mathbf{B} \notin \mathcal{H}$. By $\overline{\nabla}_\vee$, we have $\mathbf{A} \vee \mathbf{B} \notin \mathcal{H}$ and $\mathbf{A} \vee \mathbf{B} \in \mathcal{H}^{\mathtt{F}}$. Now, define $d : \mathcal{D}_o \to \mathcal{D}_{o\to o}$ by $d(\langle \mathcal{H}^{\mathsf{p}}, \mathsf{p}\rangle) := \langle \{\vee\mathbf{A} | \mathbf{A} \in \mathcal{H}^{\mathsf{p}}\}, d_{\mathsf{p}}\rangle$. It is easy to check $\langle \{\vee\}, d\rangle \in \mathcal{D}_{o\to o\to o}$. So, we let $\mathcal{E}(\vee) := \langle \{\vee\}, d\rangle$.

Finally, we must interpret the logical constants Π^α. To do so, let $\pi^\alpha : \mathcal{D}_{\alpha\to o} \to \mathcal{D}_o$ be defined by $\pi^\alpha(\langle F, f\rangle) := \langle \mathcal{H}^{\mathsf{p}}, \mathsf{p}\rangle$ where $\mathsf{p} \equiv \mathtt{T}$ if $f(\mathsf{a}) \equiv \langle \mathcal{H}^{\mathtt{T}}, \mathtt{T}\rangle$ for every $\mathsf{a} \in \mathcal{D}_\alpha$ and $\mathsf{p} \equiv \mathtt{F}$ otherwise. We need to show $\langle \{\Pi^\alpha\}, \pi^\alpha\rangle \in \mathcal{D}_{(\alpha\to o)\to o}$. Let $\langle F, f\rangle \in \mathcal{D}_{\alpha\to o}$ and $\mathbf{F} \in F$ be given. First, assume $f(\mathsf{a}) \equiv \mathtt{T}$ for every $\mathsf{a} \in \mathcal{D}_\alpha$, so that $\pi^\alpha(\langle F, f\rangle) \equiv \langle \mathcal{H}^{\mathtt{T}}, \mathtt{T}\rangle$. If $\Pi^\alpha\mathbf{F} \notin \mathcal{H}^{\mathtt{T}}$, then $\neg(\Pi^\alpha\mathbf{F}) \in \mathcal{H}$. By $\overline{\nabla}_\exists$ and $\overline{\nabla}_\beta$ (or $\overline{\nabla}_\eta$), there is a parameter $w_\alpha \in \Sigma$ such that $\neg(\mathbf{F}w){\downarrow_*} \in \mathcal{H}$. So, $(\mathbf{F}w){\downarrow_*} \notin \mathcal{H}^{\mathtt{T}}$. But this implies $f(\mathcal{E}(w)) \equiv \langle \mathcal{H}^{\mathtt{F}}, \mathtt{F}\rangle$, contradicting our assumption about f. So, $\Pi^\alpha\mathbf{F} \in \mathcal{H}^{\mathtt{T}}$. Next, assume there is an $\langle A, a\rangle \in \mathcal{D}_\alpha$ such that $f(\langle A, a\rangle) \equiv \langle \mathcal{H}^{\mathtt{F}}, \mathtt{F}\rangle$, so that $\pi^\alpha(\langle F, f\rangle) \equiv \langle \mathcal{H}^{\mathtt{F}}, \mathtt{F}\rangle$. Since A is nonempty by the definition of $\mathcal{D}_\alpha$, we can choose some $\mathbf{A} \in A$. Since $\langle F, f\rangle \in \mathcal{D}_{\alpha\to o}$, we know $(\mathbf{FA}){\downarrow_*} \in \mathcal{H}^{\mathtt{F}}$. If $\Pi^\alpha\mathbf{F} \notin \mathcal{H}^{\mathtt{F}}$, then $\Pi^\alpha\mathbf{F} \in \mathcal{H}$ and by $\overline{\nabla}_\forall$ and $\overline{\nabla}_\beta$ (or $\overline{\nabla}_\eta$), $(\mathbf{FA}){\downarrow_*} \in \mathcal{H}$, so $(\mathbf{FA}){\downarrow_*} \notin \mathcal{H}^{\mathtt{F}}$, a contradiction. So, $\Pi^\alpha\mathbf{F} \in \mathcal{H}^{\mathtt{F}}$. We let $\mathcal{E}(\Pi^\alpha) := \langle \{\Pi^\alpha\}, \pi^\alpha\rangle$.

We complete the defining of the Σ-evaluation $\mathcal{E}$ by induction on terms. On each domain $\mathcal{D}_\alpha$, let $\pi_1 : \mathcal{D}_\alpha \to \mathcal{P}(\mathit{cwff}_\alpha(\Sigma){\downarrow_*})$ be the function taking the first component of the pair, i.e., $\pi_1(\langle A, a\rangle) := A$. For each variable assignment φ, let $Sub(\varphi) := \{\theta \text{ substitution} | \forall X \in \mathcal{V}.\theta(X) \in \pi_1(\varphi(X))\}$. At each stage, we show $\theta(\mathbf{A}){\downarrow_*} \in \pi_1(\mathcal{E}_\varphi(\mathbf{A}))$ for every $\theta \in Sub(\varphi)$. Note that we defined $\mathcal{E}$ so that $c \in S$ when $\mathcal{E}(c) \equiv \langle S, s\rangle$. For variables, let $\mathcal{E}_\varphi(x) := \varphi(x)$. We defined $Sub(\varphi)$ so that $\theta \in Sub(\varphi)$ implies $\theta(x) \in \pi_1(\varphi(x))$. For application, let $\mathcal{E}_\varphi(\mathbf{F}_{\alpha\to\beta}\mathbf{A}_\alpha) := \mathcal{E}_\varphi(\mathbf{F}) @ \mathcal{E}_\varphi(\mathbf{A})$. Again, if $\theta \in Sub(\varphi)$, then $\theta(\mathbf{F}){\downarrow_*} \in \pi_1(\mathcal{E}_\varphi(\mathbf{F}))$ and $\theta(\mathbf{A}){\downarrow_*} \in \pi_1(\mathcal{E}_\varphi(\mathbf{A}))$. So, $\theta(\mathbf{FA}){\downarrow_*} \in \mathcal{E}_\varphi(\mathbf{FA})$ by the requirement on functions in the definition of $\mathcal{D}_{\alpha\to\beta}$. For abstraction, let $\mathcal{E}_\varphi(\lambda X_\alpha.\mathbf{B}_\beta) := \langle F, f\rangle$ where $F := \{\theta(\lambda X_\alpha.\mathbf{B}){\downarrow_*} | \theta \in Sub(\varphi)\}$ and $f : \mathcal{D}_\alpha \to \mathcal{D}_\beta$ is defined by $f(\mathsf{a}) := \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{B})$ for $\mathsf{a} \in \mathcal{D}_\alpha$. To see $\langle F, f\rangle \in \mathcal{D}_{\alpha\to\beta}$, we know F is $\mathcal{H}$-Boolean Compatible by Lemma 2.4.53. To check that

f is an appropriate second component, suppose $\mathsf{a} \equiv \langle A, a\rangle \in \mathcal{D}_\alpha$, $\mathbf{A} \in A$ and $\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{B}) \equiv \langle B, b\rangle$. We have $\theta((\lambda X.\mathbf{B})\mathbf{A}){\Downarrow_*} \equiv \theta([\mathbf{A}/X]\mathbf{B}){\Downarrow_*} \in B$ since $(\theta, [\theta(\mathbf{A})/X]) \in Sub(\varphi, [\mathsf{a}/X])$.

By definition, we have $\mathcal{E}_\varphi\big|_{\mathcal{V}} \equiv \varphi$ and that $\mathcal{E}$ respects application. An easy induction shows $\mathcal{E}_\varphi(\mathbf{A}) \equiv \mathcal{E}_\psi(\mathbf{A})$ whenever φ and ψ coincide on $free(\mathbf{A})$. It remains to check that $\mathcal{E}$ respects β-conversion.[28] EdNote(28)

To make this evaluation into a model, we take the obvious valuation $\upsilon : \mathcal{D}_o \to \{\mathtt{T}, \mathtt{F}\}$ defined by $\upsilon(\langle \mathcal{H}^{\mathsf{p}}, \mathsf{p}\rangle) := \mathsf{p}$. It is easy to check that, given the interpretations of the logical constants, this is a valuation, so we do have a model $\mathcal{M} := (\mathcal{D}, @, \mathcal{E}, \upsilon)$. Also, for any $\mathbf{A} \in \mathcal{H}$, we have $\mathbf{A}{\Downarrow_*} \in \mathcal{H}$ and $\mathbf{A}{\Downarrow_*} \notin \mathcal{H}^{\mathtt{F}}$. So, we must have $\mathcal{E}(\mathbf{A}) \equiv \langle \mathcal{H}^{\mathtt{T}}, \mathtt{T}\rangle$ (since $\mathbf{A}{\Downarrow_*}$ must be in the first component). That is, $\mathcal{M} \models \mathcal{H}$. Also, since $\mathcal{D}_o$ has only two elements, $\mathcal{M}$ satisfies property $\mathfrak{b}$.

In the η-case, we need to check that $\mathcal{M}$ satisfies property η. Let $F_{\alpha\to\beta}$ be a variable and φ be a variable assignment. Let $\langle F_1, f_1\rangle \equiv \varphi(F)$ and $\langle F_2, f_2\rangle \equiv \mathcal{E}_\varphi(\lambda X_\alpha.(FX))$. We need to show $F_1 \equiv F_2$ and $f_1 \equiv f_2$. By the definition of $\mathcal{E}_\varphi$, we have

$$\begin{aligned} F_2 &\equiv \{\theta(\lambda X_\alpha.(Fx)){\downarrow_{\beta\eta}} | \theta \in Sub(\psi)\} \equiv \{\theta(F){\downarrow_{\beta\eta}} | \theta \in Sub(\psi)\} \\ &\equiv \{\mathbf{F}{\downarrow_{\beta\eta}} | \mathbf{F} \in \pi_1(\psi(F))\} \equiv \{\mathbf{F}{\downarrow_{\beta\eta}} | \mathbf{F} \in F_1\} \equiv F_1 \end{aligned}$$

To check $f_1 \equiv f_2$, note that $f_2(\mathsf{a}) \equiv \mathcal{E}_{\varphi,\mathsf{a}/X}(Fx) \equiv f_1(\mathsf{a})$.

Finally, we can obtain a model satisfying property $\mathfrak{q}$ by taking the quotient as in Theorem 1.3.62. However, as in the $\mathfrak{Hint}_\beta$ and $\mathfrak{Hint}_{\beta\eta}$ case, we can show that if $\mathcal{H}$ is equation-free, then the model $\mathcal{M}$ we have constructed satisfies property $\mathfrak{q}$ already. By Theorem **??**, we can assume $\mathcal{H}$ is equation-free.

For each $\langle A, a\rangle \in \mathcal{D}_\alpha$, let $s_{\langle A,a\rangle} : \mathcal{D}_\alpha \to \mathcal{D}_o$ be defined by $s_{\langle A,a\rangle}(\langle A, a\rangle) := \langle \mathcal{H}^{\mathtt{T}}, \mathtt{T}\rangle$ and $s_{\langle A,a\rangle}(\langle B, b\rangle) := \langle \mathcal{H}^{\mathtt{F}}, \mathtt{F}\rangle$ for $\langle B, b\rangle \not\equiv \langle A, a\rangle$. This is well-defined since $\neg(\mathbf{A} \doteq \mathbf{A}'){\downarrow_*} \notin \mathcal{H}$ (by Lemma 2.4.51) implies $(\mathbf{A} \doteq \mathbf{A}'){\downarrow_*} \in \mathcal{H}^{\mathtt{T}}$ for any $\mathbf{A}, \mathbf{A}' \in A$, and $(\mathbf{A} \doteq \mathbf{B}){\downarrow_*} \notin \mathcal{H}$ (since $\mathcal{H}$ is equation-free) implies $(\mathbf{A} \doteq \mathbf{B}){\Downarrow_*} \in \mathcal{H}^{\mathtt{F}}$ for any $\mathbf{A} \in A$ and $\mathbf{B} \in B$. Then, define $\mathrm{q}^\alpha := \langle \{\doteq^\alpha\}, l\rangle$ where $l(\langle A, a\rangle) := \langle \{(\lambda X.\mathbf{A} \doteq X){\Downarrow_*} \,|\mathbf{A} \in A\}, s_{\langle A,a\rangle}\rangle$. This witnesses $\mathcal{M}$ satisfies property $\mathfrak{q}$. □

2.4.11 Proof of Model Existence Theorem

The model existence theorems we have proven above are summarized in Theorem 2.4.11. So we are now in the position to present the combined proof of Theorem 2.4.11.

Proof: Suppose we have a sufficiently Σ-pure $\Phi \in \Gamma_\Sigma$ where Γ_Σ is an acceptable abstract consistency class in $\mathfrak{Acc}_*$. Let Γ'_Σ be the compactification of Γ_Σ given by Theorem 1.6.19. This Γ'_Σ is also acceptable. Let $\mathcal{H}$ be an acceptable Hintikka set in Γ'_Σ such that $\Phi \subseteq \mathcal{H}$ guaranteed to exist by Theorem 2.4.4. We now construct a model $\mathcal{M}$ in $\mathfrak{M}_*$ such that $\mathcal{M} \models \mathcal{H}$, hence $\mathcal{M} \models \Phi$.

$\mathfrak{Acc}_\beta$,$\mathfrak{Acc}_{\beta\eta}$ These cases follow from Theorem 2.4.20.

$\mathfrak{Acc}_{\beta\mathfrak{f}}$, $\mathfrak{Acc}_{\beta\mathfrak{fb}}$ These cases follow from Theorem 2.4.39.

$\mathfrak{Acc}_{\beta\xi}$, $\mathfrak{Acc}_{\beta\xi\mathfrak{b}}$ These cases follow from Theorem 2.4.45.

$\mathfrak{Acc}_{\beta\mathfrak{b}}$,$\mathfrak{Acc}_{\beta\eta\mathfrak{b}}$ These cases follow from Theorem 2.4.54.

[28] EDNOTE: which I'll show later maybe

Draft 13/1/2003

2.5 The Saturated Extension Theorem

For each class of models $\mathfrak{M}_*$, we can produce saturated abstract consistency classes in $\mathfrak{Acc}_*$ that will be a saturated extension of all acceptable sufficiently Σ-pure Γ_Σ in $\mathfrak{Acc}_*$. Let $\Gamma_\Sigma^{\mathfrak{M}_*}$ be the class of all Φ such that there exists a model $\mathcal{M} \in \mathfrak{M}_*$ such that $\mathcal{M} \models \Phi$. We prove below that each $\Gamma_\Sigma^{\mathfrak{M}_*}$ is in $\mathfrak{Acc}_*$.

Definition 2.5.1 ($\Gamma_\Sigma^{\mathfrak{M}_*}$) We define $\Gamma_\Sigma^{\mathfrak{M}_*}$ to consist of the class of all $\Phi \in \mathit{cwff}(\Sigma)$ such that there exists a model $\mathcal{M} \in \mathfrak{M}_*$ with $\mathcal{M} \models \Phi$.

Theorem 2.5.2 *Each* $\Gamma_\Sigma^{\mathfrak{M}_*}$ *is in* $\mathfrak{Acc}_*$ *for* $* \in \{\beta, \beta\eta, \beta\xi, \beta\mathfrak{f}, \beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$.

Proof: Let $\Phi \in \Gamma_\Sigma^{\mathfrak{M}_*}$ be given and let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model in $\mathfrak{M}_*$ such that $\mathcal{M} \models \Phi$. Let $\mathbf{A}, \mathbf{B} \in \mathit{cwff}_o(\Sigma)$, $\mathbf{F}, \mathbf{G} \in \mathit{cwff}_{\alpha\to\beta}(\Sigma)$ and $\mathbf{P} \in \mathit{cwff}_{\alpha\to o}(\Sigma)$

∇_c Since $\mathcal{M}$ cannot model both $\mathbf{A}$ and $\neg\mathbf{A}$, we cannot have $\mathbf{A}, \neg\mathbf{A} \in \Phi$.

$\nabla_\neg$,∇_β, $\nabla_\vee$, $\nabla_\wedge$, $\nabla_\forall$ If $\neg\neg\mathbf{A} \in \Phi$, then $\mathcal{M} \models \Phi * \mathbf{A}$. So, $\Phi * \mathbf{A} \in \Gamma_\Sigma^{\mathfrak{M}_*}$. The other properties follow by the same argument.

$\nabla_\exists$ Suppose $\mathcal{M} \models \neg(\Pi\mathbf{P})$. Then there is some $\mathsf{a} \in \mathcal{D}_\alpha$ with $\mathcal{M} \models_\varphi \neg(\mathbf{P}X_\alpha)$ where $\varphi(X) \equiv \mathsf{a}$. Let $w \in \Sigma_\alpha$ be any constant that does not occur in Φ (note that the definition of $\nabla_\exists$ does not require that there be such a constant). By Lemma 2.6.2, there is a model $\mathcal{M}' \equiv (\mathcal{D}, @, \mathcal{E}', \upsilon) \in \mathfrak{M}_*$ such that $\mathcal{M}' \models \Phi$ (since w does not occur in any sentence in Φ) and $\mathcal{E}'(w) \equiv \mathsf{a}$. So, $\mathcal{M}' \models \neg(\mathbf{P}w)$.

∇_η We need only check this in case $* \in \{\beta\eta, \beta\eta\mathfrak{b}\}$. In this case, η directly implies $\mathcal{E}(\mathbf{A}) \equiv \mathcal{E}(\mathbf{A}{\downarrow_{\beta\eta}})$ for any $\mathbf{A} \in \mathit{cwff}_o(\Sigma)$. As a result, $\mathcal{M} \models \mathbf{A}$ iff $\mathcal{M} \models \mathbf{A}{\downarrow_{\beta\eta}}$. Suppose $\Phi \in \Gamma_\Sigma^{\mathfrak{M}_*}$, $\mathbf{A} \equiv_{\beta\eta} \mathbf{B}$ and $\mathbf{A} \in \Phi$. So, there is a model $\mathcal{M} \in \mathfrak{M}_*$ with $\mathcal{M} \models \Phi$. We have $\mathcal{M} \models \mathbf{A}$, and so $\mathcal{M} \models \mathbf{A}{\downarrow_{\beta\eta}}$ and $\mathcal{M} \models \mathbf{B}$. So, $\Phi * \mathbf{B} \in \Gamma_\Sigma^{\mathfrak{M}_*}$.

$\nabla_\mathfrak{f}$ We need only check this in case $* \in \{\beta\mathfrak{f}, \beta\mathfrak{f}\mathfrak{b}\}$. If $\mathcal{M} \models \neg(\mathbf{F} \doteq^{\alpha\to\beta} \mathbf{G})$, then we know $\mathcal{E}(\mathbf{F})$ and $\mathcal{E}(\mathbf{G})$ are distinct elements of $\mathcal{D}_{\alpha\to\beta}$[29]. Since $\mathcal{M}$ is functional, there must be an element $\mathsf{a} \in \mathcal{D}_\alpha$ with $\mathcal{E}(\mathbf{F})@\mathsf{a} \not\equiv \mathcal{E}(\mathbf{G})@\mathsf{a}$. Let X_α be a variable of type α and φ be any assignment. By property $\mathfrak{q}$, we have $\mathcal{M} \models_{\varphi,[\mathsf{a}/X]} \neg((\mathbf{F}X) \doteq^\beta (\mathbf{G}X))$. Let $w \in \Sigma_\alpha$ be any constant which does not occur in Φ. By Lemma 2.6.2, there is a model $\mathcal{M}' \equiv (\mathcal{D}, @, \mathcal{E}', \upsilon) \in \mathfrak{M}_*$ such that $\mathcal{M}' \models \Phi$ (since w does not occur in any sentence in Φ). Also, $\mathcal{M} \models_{\varphi,[\mathsf{a}/X]} (\mathbf{F}X) \doteq^\beta (\mathbf{G}X)$ implies $\mathcal{M}' \models \neg((\mathbf{F}w) \doteq^\beta (\mathbf{G}w))$ (since w does not occur in $\mathbf{F}$ or $\mathbf{G}$ and $\neg((\mathbf{F}w) \doteq^\beta (\mathbf{G}w))$ is closed). Thus, $\Phi * \neg(\mathbf{F}w \doteq^\beta \mathbf{G}w)\} \in \Gamma_\Sigma^{\mathfrak{M}_*}$. EdNote(29)

$\nabla_\mathfrak{b}$ We only must check this in case $* \in \{\beta\mathfrak{b}, \beta\eta\mathfrak{b}, \beta\xi\mathfrak{b}, \beta\mathfrak{f}\mathfrak{b}\}$. In this case $\mathcal{D}_o$ has only two elements. If $\mathcal{M} \models (\mathbf{A} \doteq^o \mathbf{B})$, then $\mathcal{E}(\mathbf{A})$ and $\mathcal{E}(\mathbf{B})$ must be distinct elements of $\mathcal{D}_o$. There are only two possibilities. We could have $\mathbf{A}$ equal to $\mathcal{E}(\mathbf{T}_o)$ (so, $\mathcal{M} \models \mathbf{A}$) and $\mathbf{B}$ equal to $\mathcal{E}(\mathbf{F}_o)$ (so, $\mathcal{M} \models \neg\mathbf{B}$). In this case, $\Phi \cup \{\mathbf{A}, \neg\mathbf{B}\} \in \Gamma_\Sigma^{\mathfrak{M}_*}$. Otherwise, we must have $\mathbf{A}$ equal to $\mathcal{E}(\mathbf{F}_o)$ (so, $\mathcal{M} \models \neg\mathbf{A}$) and $\mathbf{B}$ equal to $\mathcal{E}(\mathbf{T}_o)$ (so, $\mathcal{M} \models \mathbf{B}$). In this case, $\Phi \cup \{\neg\mathbf{A}, \mathbf{B}\} \in \Gamma_\Sigma^{\mathfrak{M}_*}$.

[29] EdNote: reference the relevant lemma here

We must finally show that each $\Gamma_\Sigma^{\mathfrak{M}_*}$ is saturated. Let $\Phi \in \Gamma_\Sigma^{\mathfrak{M}_*}$ be given and let $\mathbf{A} \in cwff_o(\Sigma)$. Let $\mathcal{M} \in \mathfrak{M}_*$ be a model of Φ. This model either satisfies $\mathbf{A}$ or $\neg\mathbf{A}$. So, $\mathcal{M}$ either witnesses $\Phi \cup \{\mathbf{A}\} \in \Gamma_\Sigma^{\mathfrak{M}_*}$ or $\Phi * \neg\mathbf{A} \in \Gamma_\Sigma^{\mathfrak{M}_*}$. Hence, $\Gamma_\Sigma^{\mathfrak{M}_*}$ is saturated. □

We are now presenting the saturated extension theorem. [30] EdNote(30)

Theorem 2.5.3 (Saturated Extension Theorem)
There is a[31] *saturated abstract consistency class Γ'_Σ in $\mathfrak{Acc}_*$ such that for every acceptable sufficiently Σ-pure abstract consistency class Γ_Σ in $\mathfrak{Acc}_*$, Γ'_Σ is a saturated extension of Γ_Σ.* EdNote(31)

Proof: Let Γ_Σ be an acceptable sufficiently Σ-pure abstract consistency class in $\mathfrak{Acc}_*$. We need only show $\Gamma_\Sigma \subseteq \Gamma_\Sigma^{\mathfrak{M}_*}$. Let $\Phi \in \Gamma_\Sigma$ be given. Since Γ_Σ is sufficiently Σ-pure, Φ is sufficiently Σ-pure. By the Model Existence Theorem 2.4.11, we have a model $\mathcal{M}$ in $\mathfrak{M}_*$ such that $\mathcal{M} \models \Phi$. This verifies $\Phi \in \Gamma_\Sigma^{\mathfrak{M}_*}$ and we are done. □ BegNP(32) EndNP(32)

2.6 Appendix

Some additional lemmata and results, which we employ need in this paper, are introduced here. [33] EdNote(33)

Lemma 2.6.1 *Let $\mathcal{A} \equiv (\mathcal{D}, @, \mathcal{E})$ be a Σ-evaluation, w_α a parameter and $\mathsf{a} \in \mathcal{D}_\alpha$. There is a Σ-evaluation $\mathcal{A}' \equiv (\mathcal{D}, @, \mathcal{E}')$ such that $\mathcal{E}'(w) \equiv \mathsf{a}$ and $\mathcal{E}'_\varphi(\mathbf{B}) \equiv \mathcal{E}_\varphi(\mathbf{B})$ for every assignment φ and $\mathbf{B} \in wff_\beta(\Sigma)$ in which w does not occur. Furthermore, if $\mathcal{E}_\varphi(\mathbf{B}) \equiv \mathcal{E}_\varphi(\mathbf{B}\!\downarrow_{\beta\eta})$ for every assignment φ and $\mathbf{B} \in wff_\beta(\Sigma)$, then $\mathcal{E}'_\varphi(\mathbf{B}) \equiv \mathcal{E}'_\varphi(\mathbf{B}\!\downarrow_{\beta\eta})$ for every assignment φ and $\mathbf{B} \in wff_\beta(\Sigma)$.*[34] EdNote(34)

Proof: We define a new evaluation function $\mathcal{E}'$ as follows. For each $\mathbf{B} \in wff_\beta(\Sigma)$, let $Y \equiv Y_\alpha^{\mathbf{B}}$ be a variable that does not occur in $\mathbf{B}$. Let $\mathbf{G} \in wff_\beta(\Sigma)$ be the unique term such that w does not occur in $\mathbf{G}$ and $\mathbf{B} \equiv [w/Y]\mathbf{G}$. Now, let $\mathcal{E}'_\varphi(\mathbf{B}) := \mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{G})$ for any assignment φ.

Note first that the definition of $\mathcal{E}'$ does not depend on the choice of variable $Y_\alpha^{\mathbf{B}}$ for each $\mathbf{B} \in wff_\beta(\Sigma)$. Let $Y \notin free(\mathbf{B})$. Let $\mathbf{G}$ and $\mathbf{G}'$ be the unique terms in which w does not occur such that $[w/Y^{\mathbf{B}}]\mathbf{G} \equiv \mathbf{B}$ and $[w/Y]\mathbf{G}' \equiv \mathbf{B}$. It is easy to show $[Y^{\mathbf{B}}/Y]\mathbf{G}' \equiv \mathbf{G}$. So, $\mathcal{E}'_\varphi(\mathbf{B}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y^{\mathbf{B}}]}(\mathbf{G}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y]}([Y/Y^{\mathbf{B}}]\mathbf{G}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{G}')$.

If $\mathbf{B} \equiv w$, then $\mathbf{G} \equiv Y^w$ and $\mathcal{E}'_\varphi(w) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y^w]}(Y^w) \equiv \mathsf{a}$. Also, if w does not occur in $\mathbf{B} \in wff_\beta(\Sigma)$, then $\mathbf{G} \equiv \mathbf{B}$. So, $\mathcal{E}'_\varphi(\mathbf{B}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{G}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{B}) \equiv \mathcal{E}_\varphi(\mathbf{B})$ where $Y \equiv Y^{\mathbf{B}} \notin free(\mathbf{B})$.

We now check that $\mathcal{E}'$ is an evaluation function by establishing the conditions in Definition 1.3.17. Let $\mathbf{F} \in wff_{\beta\to\gamma}(\Sigma)$ and $\mathbf{A} \in wff_\beta(\Sigma)$ be given, Y be any variable not free in $\mathbf{A}$, and $\mathbf{A}'$ be the unique term in which w does not occur and $[w/Y]\mathbf{A}' \equiv \mathbf{A}$.

1. Since w does not occur in any variable X_β, we have $\mathcal{E}'_\varphi(X) \equiv \mathcal{E}_\varphi(X) \equiv \varphi(X)$.

[30] EdNote: discuss the relevance of the $\exists\forall$ result, which is surprising; one would have expected a $\forall\exists$ result. What are the consequences.

[31] EdNote: sufficiently Σ-pure?

[32] New Part: no change, but do we need to have this so early? Chris: In my opinion we should give the proof here. Otherwise we will jump forth and back in the paper. But if you guys decide that it should come later in paper, then it is also fine with me.

[33] EdNote: Chris: I refer to this lemmata already in the Soundness argument of the Sequent calculi. Actually these lemmata should go in the other paper (I guess they are even of relevance thjere too) and we could do this in the final jsl version. Alternatively we could completely skip them and leave them to the reader???

[34] EdNote: we can reformulate the last sentence as: if $\mathcal{A}$ satisfies η, then $\mathcal{A}'$ does too. Chris: yes we should do.

2. Let $\mathbf{A} \equiv \mathbf{FB}$, then $\mathbf{A}' \equiv \mathbf{F}'\mathbf{B}'$. For any assignment φ, we have

$$\mathcal{E}'_\varphi(\mathbf{FB}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{F}'\mathbf{B}') \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{F}')@\mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{B}') \equiv \mathcal{E}_\varphi(\mathbf{F})@\mathcal{E}_\varphi(\mathbf{B})$$

3. Let φ and ψ are assignments which coincide on $free(\mathbf{A})$. Since $\varphi, [\mathsf{a}/Y]$ and $\psi, [\mathsf{a}/Y]$ coincide on $free(\mathbf{A}')$, we have $\mathcal{E}'_\varphi(\mathbf{A}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{A}') \equiv \mathcal{E}_{\psi,[\mathsf{a}/Y]}(\mathbf{A}') \equiv \mathcal{E}'_\psi(\mathbf{A})$.

4. Note that Y also is not free in $\mathbf{A}\!\downarrow_\beta$, w does not occur in $\mathbf{A}'\!\downarrow_\beta$, and $[w/Y]\mathbf{A}'\!\downarrow_\beta \equiv ([w/Y]\mathbf{A}')\!\downarrow_\beta$. So, $\mathbf{A}'\!\downarrow_\beta$ is the unique term in which w does not occur and $[w/Y]\mathbf{A}'\!\downarrow_\beta \equiv ([w/Y]\mathbf{A}')\!\downarrow_\beta$. We compute $\mathcal{E}'_\varphi(\mathbf{A}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{A}') \equiv \mathcal{E}_{\varphi,[\mathsf{a}/Y]}(\mathbf{A}'\!\downarrow_\beta) \equiv \mathcal{E}'_\varphi(\mathbf{A}\!\downarrow_\beta)$.

The same argument shows $\mathcal{E}'_\varphi(\mathbf{A}) \equiv \mathcal{E}'_\varphi(\mathbf{A}\!\downarrow_{\beta\eta})$ if $\mathcal{E}$ respects $\beta\eta$-normalization. ☐

Lemma 2.6.2 *Let $\mathcal{M} \equiv (\mathcal{D}, @, \mathcal{E}, \upsilon)$ be a Σ-model, w_α a parameter, and $\mathsf{a} \in \mathcal{D}_\alpha$. Then there is a Σ-model $\mathcal{M}' \equiv (\mathcal{D}, @, \mathcal{E}', \upsilon)$, such that*

1. *For every $\mathbf{C} \in wff_o(\Sigma)$ in which w does not occur, assignment φ, and variable X_α, $\mathcal{M}' \models_\varphi [w/X]\mathbf{C}$ whenever $\mathcal{M} \models_{\varphi,[\mathsf{a}/X]} \mathbf{C}$.*
2. *If $\mathbf{C}$ is closed and w does not occur in $\mathbf{C}$, then $\mathcal{M}' \models \mathbf{C}$ whenever $\mathcal{M} \models \mathbf{C}$.*
3. *If $\mathcal{M} \in \mathfrak{M}_*$, then $\mathcal{M}' \in \mathfrak{M}_*$.*

Proof: To establish the first assertion let $(\mathcal{D}, @, \mathcal{E}')$ be the evaluation for $(\mathcal{D}, @, \mathcal{E})$ given by Lemma 2.6.1. To know $\mathcal{M}'$ is a Σ-model, we need only check that υ is a valuation for $(\mathcal{D}, @, \mathcal{E}')$. All the conditions follow trivially from the fact that $\mathcal{E}'(\neg) \equiv \mathcal{E}(\neg)$, $\mathcal{E}'(\vee) \equiv \mathcal{E}(\vee)$, and $\mathcal{E}'(\Pi^\beta) \equiv \mathcal{E}(\Pi^\beta)$ for each type β. If $\mathcal{M}$ satisfies property $\mathfrak{q}$, then $\mathcal{M}'$ satisfies property $\mathfrak{q}$ since the definition of property $\mathfrak{q}$ does not use the evaluation function.

Let $\mathbf{C} \in wff_o(\Sigma)$ in which w does not occur, an assignment φ, and a variable X_α be given. Suppose $\mathcal{M} \models_{\varphi,[\mathsf{a}/X]} \mathbf{C}$. That is, $\upsilon(\mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{C})) \equiv \mathtt{T}$. Since w does not occur in $\mathbf{C}$, we have $\mathcal{E}'_{\varphi,[\mathsf{a}/X]}(\mathbf{C}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}(\mathbf{C})$. Using the substitution property (for $\mathcal{E}'$), we have $\mathcal{E}'_\varphi([w/X]\mathbf{C}) \equiv \mathcal{E}'_{\varphi,[\mathsf{a}/X]}(\mathbf{C})$ since $\mathcal{E}'(w) \equiv \mathsf{a}$. So, $\mathcal{E}'_\varphi([w/X]\mathbf{C}) \equiv \mathcal{E}_{\varphi,[\mathsf{a}/X]}\mathbf{C}$ and hence $\upsilon(\mathcal{E}'_\varphi([w/X]\mathbf{C})) \equiv \upsilon(\mathcal{E}_{\varphi,[\mathsf{a}/X]}\mathbf{C}) \equiv \mathtt{T}$. That is, $\mathcal{M}' \models_\varphi [w/X]\mathbf{C}$.

The second assertion is a trivial consequence of the first one, and to see the third assertion we proceed as for $\mathfrak{q}$ in the first one: whether properties $\mathfrak{b}$ and $\mathfrak{f}$ hold is determined by the underlying applicative structure, which is the same for $\mathcal{M}$ and $\mathcal{M}'$. Finally, if $\mathcal{M}$ satisfies property η, then we know $\mathcal{M}'$ satisfies property η by Lemma 2.6.1. ☐

2.7 Conclusion

Acknowledgments

The work presented in this paper has been supported by the "Deutsche Forschungsgemeinschaft" (DFG) under Grant HOTEL, the National Science Foundation under Grant CCR-0097179 and a Heisenberg stipend to the second author. The authors would like to thank Peter Andrews and Frank Pfenning for stimulating discussions.

Bibliography

[ABB00] Peter B. Andrews, Matthew Bishop, and Chad E. Brown. TPS: A theorem proving system for type theory. In David McAllester, editor, *Proceedings of the 17th International Conference on Automated Deduction*, number 1831 in LNAI, pages 164–169, Pittsburgh, USA, 2000. Springer.

[ABI+96] Peter B. Andrews, Matthew Bishop, Sunil Issar, Dan Nesmith, Frank Pfenning, and Hongwei Xi. TPS: A theorem proving system for classical type theory. *Journal of Automated Reasoning*, 16(3):321–353, 1996.

[And71] Peter B. Andrews. Resolution in type theory. *Journal of Symbolic Logic*, 36(3):414–432, 1971.

[And72a] Peter B. Andrews. General models and extensionality. *Journal of Symbolic Logic*, 37(2):395–397, 1972.

[And72b] Peter B. Andrews. General models descriptions and choice in type theory. *Journal of Symbolic Logic*, 37(2):385–394, 1972.

[And73] Peter B. Andrews, 1973. letter to Roger Hindley dated January 22, 1973.

[And89] Peter B. Andrews. On Connections and Higher Order Logic. *Journal of Automated Reasoning*, 5:257–291, 1989.

[And02] Peter B. Andrews. *An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof.* Kluwer Academic Publishers, second edition, 2002.

[Bar84] Henk P. Barendregt. *The Lambda Calculus.* North Holland, 1984.

[BBK02a] Christoph Benzmüller, , Chad E. Brown, and Michael Kohlhase. Model existence for higher-order logic. submitted to JSL, 2002.

[BBK02b] Christoph Benzmüller, Chad E. Brown, and Michael Kohlhase. Semantic techniques for higher-order cut-elimination. manuscript, http://www.ags.uni-sb.de/ chris/papers/R23.pdf, 2002.

[BBK03] Christoph Benzmüller, , Chad E. Brown, and Michael Kohlhase. Model existence for higher-order logic. Technical Report CMU-01-03, Matematics Department, Carnegie Mellon University, 2003.

[Ben99a] Christoph Benzmüller. *Equality and Extensionality in Automated Higher-Order Theorem Proving.* PhD thesis, Saarland University, 1999.

[Ben99b] Christoph Benzmüller. Extensional higher-order paramodulation and RUE-resolution. In Harald Ganzinger, editor, *Proceedings of the 16th International Conference on Automated Deduction*, number 1632 in LNAI, pages 399–413, Trento, Italy, 1999. Springer.

[BK97] Christoph Benzmüller and Michael Kohlhase. Model existence for higher-order logic. SEKI-Report SR-97-09, Saarland University, 1997.

[BK98] Christoph Benzmüller and Michael Kohlhase. Extensional higher order resolution. In Claude Kirchner and Hélène Kirchner, editors, *Proceedings of the 15th Conference on Automated Deduction*, number 1421 in LNAI, pages 56–72, Lindau, Germany, 1998. Springer.

[BS98] Wolfgang Bibel and Peter Schmitt, editors. *Automated Deduction – A Basis for Applications.* Kluwer, 1998.

[Chu40] Alonzo Church. A formulation of the simple theory of types. *Journal of Symbolic Logic*, 5:56–68, 1940.

[CT88] Gennaro Chierchia and Raymond Turner. Semantics and property theory. *Linguistics and Philosophy*, 11:261–302, 1988.

[dB72] Nicolaas Govert de Bruijn. Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with an application to the Church-Rosser theorem. *Indagationes Mathematicae*, 34(5):381–392, 1972.

[DeM99] Mary DeMarco. *Intuitionistic Semantics for Heriditarily Harrop Logic Programming.* PhD thesis, Wesleyan University, 1999.

[DHK01] Gilles Dowek, Thérèse Hardin, and Claude Kirchner. HOL-$\lambda\sigma$ an intentional first-order expression of higher-order logic. *Mathematical Structures in Computer Science*, 11(1):1–25, 2001.

[Fit96] Melvin Fitting. *First-Order Logic and Automated Theorem Proving, second edition.* Graduate Texts in Computer Science. Springer, 1996.

[Fit02] Melvin Fitting. *Types, Tableaus, and Gödel's God.* Kluwer, 2002.

[GM93] M. J. C. Gordon and T. F. Melham. *Introduction to HOL – A theorem proving environment for higher order logic.* Cambridge University Press, 1993.

[Göd31] Kurt Gödel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte der Mathematischen Physik*, 38:173–198, 1931. English Version in [vH67].

[Hen50] Leon Henkin. Completeness in the theory of types. *Journal of Symbolic Logic*, 15(2):81–91, 1950.

[Hen96] Leon Henkin. The discovery of my completeness proofs. *The Bulletin of Symbolic Logic*, 2(2):127–158, 1996.

[Hin55] K. J. J. Hintikka. Form and content in quantification theory. *Acta Philosophica Fennica*, 8:7–55, 1955.

[HL99] Furio Honsell and Marina Lenisa. Coinductive characterizations of applicative structures. *Mathematical Structures in Computer Science*, 9:403–435, 1999.

[HS86] Roger J. Hindley and Jonathan P. Seldin. *Introduction to Combinators and Lambda-Calculs.* Cambridge University Press, Cambridge, 1986.

[HS99] Furio Honsell and Donald Sannella. Pre-logical relations. In *Proceedings of Computer Science Logic (CSL'99)*, volume 1683 of *LNCS*, pages 546–561. Springer, 1999.

[Hue72] Gérard P. Huet. *Constrained Resolution: A Complete Method for Higher Order Logic.* PhD thesis, Case Western Reserve University, 1972.

[Hue73] Gérard P. Huet. A mechanization of type theory. In Donald E. Walker and Lewis Norton, editors, *Proceedings of the 3rd International Joint Conference on Artificial Intelligence*, pages 139–146, 1973.

[JP72] D. C. Jensen and Thomasz Pietrzykowski. A complete mechanization of (ω)-order type theory. In *Proceedings of the ACM Annual Conference*, volume 1, pages 82–92, 1972.

[Koh94] Michael Kohlhase. *A Mechanization of Sorted Higher-Order Logic Based on the Resolution Principle.* PhD thesis, Saarland University, 1994.

[Koh95] Michael Kohlhase. Higher-order tableaux. In Peter Baumgartner, Reiner Hähnle, and Joachim Posegga, editors, *Theorem Proving with Analytic Tableaux and Related Methods*, number 918 in LNAI, pages 294–309, 1995.

[KS99] Michael Kohlhase and Ortwin Scheja. Higher-order multi-valued resolution. *Journal of Applied Non-Classical Logics*, 9(4), 1999.

[LP00] Shalom Lappin and Carl Pollard. Strategies for hyperintensional semantics. Manuscript, King's College, London and Ohio State University, 2000.

[LP02] Shalom Lappin and Carl Pollard. A higher-order fine-grained logic for intensional semantics. Manuscript, 2002.

[LS95] Richard Larson and Gabriel Segal. *Knowledge of Meaning.* MIT Press, Cambridge Ma, 1995.

[Mil83] Dale Miller. *Proofs in Higher-Order Logic.* PhD thesis, Carnegie-Mellon University, 1983.

[Mil91] Dale Miller. A logic programming language with lambda-abstraction, function variables, and simple unification. *Journal of Logic and Computation*, 4(1):497–536, 1991.

[Mit96] John C. Mitchell. *Foundations for Programming Languages.* Foundations of Computing. MIT Press, 1996.

[NM94] Gopalan Nadathur and Dale Miller. Higher-order logic programming. Technical Report CS-1994-38, Department of Computer Science, Duke University, 1994.

[NPW02] Tobias Nipkow, Lawrence C. Paulson, and Markus Wenzel. *Isabelle/HOL — A Proof Assistant for Higher-Order Logic.* Number 2283 in LNCS. Springer, 2002.

[RV01] J. Alan Robinson and Andrei Voronkov. *Handbook of Automated Reasoning.* MIT Press, 2001.

[Sch60] Kurt Schütte. Semantical and syntactical properties of simple type theory. *Journal of Symbolic Logic*, 25:305–326, 1960.

[Sch02] Lutz Schröder. Henkin models for the partial λ-calculus. Manuscript, cf. `http://www.informatik.uni-bremen.de/~lschrode/hascasl/henkin.ps`, 2002.

[Sea02] Jörg Siekmann and Christoph Benzmüller et al. Proof development with OMEGA. In Andrei Voronkov, editor, *Proceedings of the 19th Conference on Automated Deduction*, number 2392 in LNAI, pages 143–148, Copenhagen, Denmark, 2002. Springer.

[SM02] L. Schröder and T. Mossakowski. Hascasl: towards integrated specification and development of functional programs. In *Algebraic Methodology And Software Technology*, volume 2422 of *LNCS*, pages 99–116. Springer, 2002.

[Smu63] Raymond M. Smullyan. A unifying principle for quantification theory. *Proc. Nat. Acad Sciences*, 49:828–832, 1963.

[Smu68] Raymond M. Smullyan. *First-Order Logic.* Springer, 1968.

[Tak68] Moto-o Takahashi. Cut-elimination in simple type theory with extensionality. *Journal of the Mathematical Society of Japan*, 19, 1968.

[Tak87] Gaisi Takeuti. *Proof Theory.* North Holland, 1987.

[Tom80] R. Tomason. A model theory for proposistional attitudes. *Linguistics and Philosophy*, 4:47–70, 1980.

[vH67] Jean van Heijenoort. *From Frege to Gödel : a source book in mathematical logic 1879-1931.* Source books in the history of the sciences series. Harvard Univ. Press, Cambridge, MA, 3rd printing, 1997 edition, 1967.

[Wol94] David A. Wolfram. A semantics for λ-PROLOG. *Theoretical Computer Science*, 136(1), 1994.