School of Computer Science and Engineering

Faculty of Engineering

The University of New South Wales

Extensionality Translations for Polymorphic Higher-Order Logic

by

Vincent Jackson

Honours Thesis Report submitted as a requirement for the degree of Bachelor of Science (Honours)

Submitted: May 2020 Student ID: z5060345

Supervisor: Dr. Christine Rizkallah

Abstract

Translations between logics have been studied for almost as long a modern logic itself, the standard example being Glivenko's 1929 translation from classical to intuitionistic propositional logic. Translations show how to convert proofs in one logical system to proofs in another, demonstrating an embedding between these logics. They are thus an important tool in understanding the relationship between different logics. In the study of higher-order logics, there are several results [Gandy, 1956; Rizkallah, 2009] detailing the translation from logics with function-extensionality to those without. This thesis extends these translation results to the elimination of function-extensionality in a higher-order logic with type-polymorphism, similar to HOL2P [Völker, 2007].

Contents

Chapter 1

Introduction

1.0⟨1⟩ This thesis outlines a translation of theorems from an function-extensional to a function-intensional higher-order logic both with type-level polymorphism, by removing the axioms that imply functional extensionality in the translation.

1.0⟨2⟩ Functional extensionality is a property of equality that says if two functions have in equal outputs for every equal inputs, the functions are equal. Adding functional extensionality to a logic prevents the expression of more fine properties of functions which are dependent on how they are defined, such as run-time and structural properties.

1.0⟨3⟩ There are many translations that remove functional extensionality from higher-order logic [Gandy, 1956; Luckhardt, 1973; Rizkallah, 2009]. These translations map the extensional equality of the extensional logic into an extensional equivalence in the intensional logic, which is weaker than the equality of the target logic, but that acts like extensional equality under certain assumptions.

1.0⟨4⟩ This report extends previous function-extensionality translation results to a higher-order logic with type-level polymorphism, based on the HOL2P logic of Völker [2007]. This entails dealing with a number of details, such as how to deal with abstract type-variables, which have not been addressed in previous translations.

1.0⟨5⟩ Chapter 2 outlines the foundations of this research, giving an in-depth definition of translations of logics, higher-order logic, and function extensionality. Chapter 3 contains a summary of the previous work into extensionality translations in higher-order logic, describing previous translations which remove function-extensionality in higher-order logic. Chapter 4 explains the translation result, containg a prose description of the proofs. Chapter 5 gives the conclusions from this research.

1.0⟨6⟩ The appendices contain detailed natural deduction proofs of the translation. Appendix A contains the fundamental definitions and lemmas about the polymorphic higher-order logics. Appendix B contains fundamental definitions and lemmas necessary to perform the translation. Appendix C contains proofs of the correctness of the translation.

Chapter 2

Foundations

2.0⟨1⟩ The purpose of this chapter is to provide an overview of the topic of this thesis—the translation of extensionality in polymorphic higher order logic. To understand this we will first need to understand three things: what a translation is, what functional extensionality is, and what polymorphic higher-order logic is.

2.0⟨2⟩ To begin, we will discuss a general description of logical translations. Then, starting with a description of the polymorphic lambda calculus, System F, develop a polymorphic HOL based on Völker's HOL2P [Völker, 2007]. Then we will discuss functional extensionality, and use this understanding to produce an intensional version of polymorphic HOL.

2.1 Meta-logic and Object-logic

2.1⟨1⟩ Before we begin, it is best to clarify the conventions we shall use for our logic and meta-logic. When describing a logic, there is a difference between the logic which we describe, and the logic we describe it in; these are the *logic* and *meta-logic* respectively.

2.1⟨2⟩ We will use the symbols $\equiv$ for meta-level equality and $\Longrightarrow$ for meta-level implication. The notion of object-logic provability we will use is syntactic consequence in a natural deduction system, where $\vdash$; to elaborate, $\Gamma \vdash p$ has the general meaning "assuming the formulas in the (multi)set of formulas Γ, p is derivable", by rules which manipulate the formula p. Lastly, $\Gamma \vdash^* \Delta$ is an extension of the $\vdash$ syntax to mean that each formula in Δ can be proven from Γ.

2.1⟨3⟩ Especially when dealing with theorems of the object-logic, we will write meta-level implication in Gentzen notation. In Gentzen notation, an implication $A_1 \Longrightarrow A_2 \Longrightarrow \cdots \Longrightarrow A_n \Longrightarrow C$ is written

$$\frac{A_1 \quad A_2 \quad \cdots \quad A_n}{C}$$

When there are no assumptions, as in

$$\frac{}{C}$$

this signifies that C is true or derivable. As a convention of speech, this report uses "hypothesis" and "conclusion" to refer to the assumptions and conclusions of the meta-logical arrow ($\Longrightarrow$), and "antecedents" and "consequent" to refer to the assumed formulae and proved formula of syntactic consequence ($\vdash$).

2.1⟨4⟩ Gentzen inferences can be chained to form a proof tree. Consider the rule *trans*,

$$\frac{\Gamma \vdash s = t \qquad \Delta \vdash t = u}{\Gamma, \Delta \vdash s = u}\ \text{trans}$$

which describes the transitivity of equality. To show the proof of $s = v$ from the antecedents $s = t$, $t = u$, and $u = v$, we can write the following proof-tree

$$\dfrac{\dfrac{}{s = t \vdash s = t}\ \text{assm} \qquad \dfrac{\dfrac{}{t = u \vdash t = u}\ \text{assm} \qquad \dfrac{}{u = v \vdash u = v}\ \text{assm}}{t = u, u = v \vdash t = u}\ \text{trans}}{s = t, t = u, u = v \vdash s = v}\ \text{trans}$$

2.1⟨5⟩ In addition to these general conventions, we will define some abbreviations to simplify long derivations. If a rule is applied repeatedly, such as trans in the last example, we may denote this using the syntax $^+$ on the rule-name to denote repeated application, or * to represent zero-or-many applications (this is used when such applications depend on the presence of antecedents which may not be present in all derivations). Using this to simplify the previous derivation would result in

$$\dfrac{\dfrac{}{s = t \vdash s = t}\ \text{assm} \qquad \dfrac{}{t = u \vdash t = u}\ \text{assm} \qquad \dfrac{}{u = v \vdash u = v}\ \text{assm}}{s = t, t = u, u = v \vdash s = v}\ \text{trans}^+$$

2.2 Logical Translations

2.2⟨1⟩ A logical translation is a mapping of theorems between two systems of logic. To be more precise, given two systems of logic, *System S* (for source) and *System T* (for target), a logical translation maps all true theorems of *System S* to true theorems in *System T*. Moreover, to prevent trivial transformations such as mapping every formula from *System S* to 'True', we also require that any theorem whose translation is provable in *System T* must be provable in *System S*.

2.2⟨2⟩ If we consider theorems as natural deduction sequents, we can represent any general logical translation in the following manner. A translation between logical systems can be

considered as two functions $\mathcal{A} : \mathcal{P}(\text{Term}) \times \text{Term} \to \mathcal{P}(\text{Term})$, which generates a set of translated antecedents, and $\mathcal{C} : \mathcal{P}(\text{Term}) \times \text{Term} \to \text{Term}$, which generates a translated consequent. The reason for such generality in these functions is that we may want the translated antecedents or consequent to depend on the original antecedents in a more complex manner than simply translating each antecedents and consequent term individually.

2.2⟨3⟩ The translation must satisfy the following conditions:

$$\Gamma \vdash_S s \implies \mathcal{A}(\Gamma, p) \vdash_T \mathcal{C}(\Gamma, p) \qquad \text{(Preserves Validity)}$$
$$\mathcal{A}(\Gamma, p) \vdash_T \mathcal{C}(\Gamma, p) \implies \Gamma \vdash_S s \qquad \text{(Not Trivial)}$$

2.2⟨4⟩ As this is a very abstract definition, it is helpful to consider a concrete example. We will specify Gödel's translation [Gödel, 1933] between the classical and intuitionistic propositional logic in this framework. This translation takes theorems of classical propositional logic to theorems in intuitionistic propositional logic (that is, the propositional logic without the law of excluded middle).

2.2⟨5⟩ To begin, we define a mapping $(-)^\bullet$ on propositional formulas as follows

$$\begin{aligned} x^\bullet &\triangleq x \\ (p \to q)^\bullet &\triangleq \neg(p^\bullet \wedge \neg q^\bullet) \\ (p \vee q)^\bullet &\triangleq \neg(\neg p^\bullet \wedge \neg q^\bullet) \\ (p \wedge q)^\bullet &\triangleq p^\bullet \wedge q^\bullet \\ (\neg p)^\bullet &\triangleq \neg p^\bullet \end{aligned}$$

In addition, we need to extend $(-)^\bullet$ to sets of formulae by applying it to each element of the set.

2.2⟨6⟩ We can then define Gödel's translation in terms of our translation framework as

$$\begin{aligned} \mathcal{A}(\Gamma, p) &\triangleq \Gamma^\bullet \\ \mathcal{C}(\Gamma, p) &\triangleq p^\bullet \end{aligned}$$

That this translation preserves validity is a theorem of Gödel's [1933], and non-triviality follows from the fact that $(-)^\bullet$ maps formulae into classically equivalent formulae.

2.3 The Polymorphic Lambda Calculus: System F

2.3⟨1⟩ To discuss functional extensionality, we must first have a way to discuss functions. In computer science and logic, one of the main theoretical frameworks for describing functions is the lambda calculus. In particular, as we will be considering a polymorphic HOL, we shall consider the polymorphic lambda calculus: System F. For a more in-depth discussion of the lambda-calculus, see Barendregt's "The Lambda Calculus: Its Syntax and Semantics" [Barendregt, 1981], and for System F, Pierce's "Types and Programming Languages" [Pierce, 2002].

2.3.1 Types and Terms

2.3⟨2⟩ System F is an extension to the lambda calculus which adds ***types***, which can be constructed in three basic ways:

$$\begin{aligned} \tau, \sigma ::= \alpha & \qquad \text{(type variables)} \\ \mid \tau \Rightarrow \sigma & \qquad \text{(functions)} \\ \mid \Pi\alpha.\ s & \qquad \text{(polymorphic functions).} \end{aligned}$$

2.3⟨3⟩ Type variables (we will use α, β, and γ to stand for type-variables) represent a type about which which we know nothing, except that it may—in future—be substituted for by a more concrete type. Function types are written $\tau \Rightarrow \sigma$, which represents the type of a function term which takes a term of type τ and returns a term of type σ. Lastly, the type of polymorphic functions $\Pi\alpha.\ \tau$, is the type of a function which is abstracted over some type, represented by the variable α, and which returns a type τ, which may contain α.

2.3⟨4⟩ In System F, there are five basic ways to form *terms*

$$\begin{aligned} s, t ::= x_\tau & \qquad \text{(variables)} \\ \mid \lambda x_\tau.\ s & \qquad \text{(lambda abstraction)} \\ \mid s\ t & \qquad \text{(application)} \\ \mid \Lambda\alpha.\ s & \qquad \text{(type abstraction)} \\ \mid s\ [\!:\tau:\!] & \qquad \text{(type application).} \end{aligned}$$

Variables (we will use x, y, and z to stand for variables) represent a variable, annotated with their type. It should be noted that the variable x_τ is distinct from x_σ when $\tau \not\equiv \sigma$. $\lambda x_\tau.\ s$ represents a function which takes some input x of type τ and then proceeds with the computation s. Application applies a term t to a function s. The polymorphic function $\Lambda\alpha.\ s$ takes some type α, and proceeds with the computation s. Lastly, $s\ [\!:\tau:\!]$ represents the application of a type to a polymorphic function.

2.3⟨5⟩ For convenience, we define the following abbreviations

$$\begin{gathered} \lambda x_\tau.\ \lambda y_\sigma.\ s \rightsquigarrow \lambda x_\tau\ y_\sigma.\ s \\ \Lambda\alpha.\ \Lambda\beta.\ s \rightsquigarrow \Lambda\alpha\ \beta.\ s \\ (s\ t)\ u \rightsquigarrow s\ t\ u \\ (s\ [\!:\tau:\!])\ [\!:\sigma:\!] \rightsquigarrow s\ [\!:\tau:\!]\ [\!:\sigma:\!] \\ (s\ [\!:\tau:\!])\ u \rightsquigarrow s\ [\!:\tau:\!]\ u \\ (s\ t)\ [\!:\sigma:\!] \rightsquigarrow s\ t\ [\!:\sigma:\!] \end{gathered}$$

2.3⟨6⟩ Types are associated with terms according to a set of typing rules (Figure 2.1). In short λ-abstractions are typed to functions, and Λ-abstractions are typed to polymorphic functions. An application $(s\ t)$ passes the argument t to a function s, which results in a term of the return type of that function. Finally a type-application $s\ [\!:\sigma:\!]$ applies σ to a polymorphic function s, and results in a term of a type specialised to the input type: $s[\sigma/\alpha]$.

$$\frac{}{\vdash x_\tau : \tau} \qquad \frac{\vdash s : \tau_2}{\vdash \lambda x_{\tau_1}.\ s : \tau_1 \Rightarrow \tau_2} \qquad \frac{\vdash s : \tau_1 \Rightarrow \tau_2 \quad \vdash t : \tau_1}{\vdash s\ t : \tau_2}$$

$$\frac{\vdash s : \tau}{\vdash \Lambda\alpha.\ s : \Pi\alpha.\ \tau} \qquad \frac{\vdash s : \Pi\alpha.\ \tau}{\vdash s\ [\!:\sigma:\!] : \tau[\sigma/\alpha]}$$

Figure 2.1: The Typing Rules of System F

2.3.2 Free and Bound Variables

2.3⟨7⟩ An important aspect of variables is whether they are bound or free. We say that a variable is ***bound*** when it occurs under a lambda that binds that variable; for example x_τ is bound in $\lambda x_\tau.\ x_\tau$, and x, y, and α are bound in $\Lambda\alpha.\ \lambda y_\alpha\ x_{\alpha\Rightarrow\alpha}.\ y_\alpha\ x_{\alpha\Rightarrow\alpha}$, Contrarily, a variable is ***free*** if it does ***not*** occur under a lambda that binds it, For example y_τ is free in $y_\tau\ (\lambda x_\tau.\ x_\tau)$, and x_τ is bound and y_τ is free in $\lambda x_\tau.\ x_\tau\ y_\tau$. The meta-logic function $\mathrm{FV}(-)$ takes a term to the set of the free variables in it; the meta-logic function $\mathrm{FV}_{\mathrm{ty}}(-)$ takes a type to the set of the free variables in it. Moreover, we can extend $\mathrm{FV}_{\mathrm{ty}}(-)$ to terms by taking all free variables in all types occurring in that term (including on variables).

2.3⟨8⟩ A variable that is free is considered to be different to bound variables with the same name. For example, the lambda term $x_\tau\ (\lambda x_\tau.\ x_\tau)$ should be read $x_\tau\ (\lambda \underline{x_\tau}.\ \underline{x_\tau})$. Moreover, variables that are bound multiple times are associated with the last lambda binder under which they appear; for example $\lambda x_\tau.\ (\lambda x_\tau.\ (\lambda x_\tau.\ x_\tau)\ x_\tau)\ x_\tau$ should be read as $\lambda x_\tau.\ (\lambda \underline{x_\tau}.\ (\lambda \underline{\underline{x_\tau}}.\ \underline{\underline{x_\tau}})\ \underline{x_\tau})\ x_\tau$.

2.3⟨9⟩ As ascribing every variable with a type is tiresome and occasionally distracting, we will write bare variables x when their types do not matter, or can be ascertained from the context. In these cases, we will always assume a bound x has the ***same type*** as any other bound x (according to the conventions discussed above), and a free x has the same type as any other free x. To clarify with an example, the bare term $\lambda x.\ x\ (\lambda x.\ y\ x)$ should be read as $\lambda x_{\sigma\Rightarrow\tau_2}.\ x_{\sigma\Rightarrow\tau_2}\ (\lambda x_\tau.\ y_{\tau\Rightarrow\sigma}\ x_\tau)$.

2.3.3 Equivalence of Lambda Terms

2.3⟨10⟩ The notion of ***substitution*** of a variable is critical to the lambda calculus. In System F, there are two notions of substitution, substitution of variables and substitution of type variables.

2.3⟨11⟩ Substitution of ***variables*** is written $[s/x]$, and represents the capture-avoiding substitution of the term s for the variable x (where s and x agree on their type). Capture-avoiding means that $(\lambda x.\ s)[t/x]$ will result in the term $(\lambda x.\ s\ x)$, and ***not*** $(\lambda x.\ s\ t)$. We will abstract

over repeated term-substitution with the letter θ—e.g. $\theta \equiv (s_1/x_1, s_2/x_2, \ldots, s_n/x_n)$.

2.3⟨12⟩ Substitution of ***type-variables*** is written $[\tau/\alpha]$, and represents the capture-avoiding substitution of the type τ for the type-variable α. On ***types***, a type substitution replaces free type variables with the given type, respecting the Π binder. On ***terms***, type substitution changes variables, turning x_τ into $x_{\tau[\sigma/\alpha]}$, but respecting the binder Λ. We will abstract over repeated type-substitution with the letter ϕ—e.g. $\phi \equiv (\sigma_1/\alpha_1, \sigma_2/\alpha_2, \ldots, \sigma_n/\alpha_n)$.

Alpha Equivalence

2.3⟨13⟩ A ***change in bound variables*** of a term s is the replacement of a subterm of the form $\lambda x.\ t$ or $\Lambda\alpha.\ t$, with $\lambda y.\ t[y/x]$ or $\Lambda\beta.\ t[\beta/\alpha]$ respectively, where y or α does not occur ***at all*** in in t. Moreover, a change in bound variables can also be applied to polymorphic-function types, rewriting $\Pi\alpha.\ \tau$ to $\Pi\beta.\ \tau[\beta/\alpha]$.

2.3⟨14⟩ We say that two terms (or two types) are ***alpha equivalent*** when either of the terms can be transformed into the other by a sequence of changes of bound variables. For example $(\lambda x\ y.\ x) \equiv (\lambda y\ x.\ y)$ and $(\Lambda\alpha.\ x_\alpha) \equiv (\Lambda\beta.\ x_\beta)$, but $(\lambda x.\ x\ y) \not\equiv (\lambda y.\ y\ y)$. Note that $x\ (\lambda z.\ z)$ is ***not*** alpha equivalent to $y\ (\lambda z.\ z)$, as x and y are ***free***. From this point, we will consider all terms and types up to alpha equivalence.

Beta Equivalence

2.3⟨15⟩ The next fundamental concept in the lambda calculus is beta reduction. A ***beta reduction*** of a term s to a term s' is the replacement of some subterm of the form $(\lambda x.\ u)\ v$ with $u[v/x]$, or some subterm of the form $(\Lambda\alpha.\ u)\ [:\tau:]$ with $u[\tau/\alpha]$. For example, a beta-reduction of the term $(\Lambda\alpha.\ (\lambda x_\alpha.\ x_\alpha))\ [:\tau:]\ s$ is as follows

$$\begin{aligned}(\Lambda\alpha.\ (\lambda x_\alpha.\ x_\alpha))\ [:\tau:]\ s &\equiv (\lambda x_\tau.\ x_\tau)\ s \\ &\equiv s\end{aligned}$$

2.3⟨16⟩ A term with no beta-reducts is said to be in ***normal form***. It is widely-known that beta reduction of System F is ***confluent***—that is, all terms eventually reduce to a normal form, and any sequence of beta reductions will eventually arrive at that same normal form [Sørensen and Urzyczyn, 1998].

2.4 Extensional Polymorphic Higher-Order Logic

2.4⟨1⟩ The first logic we will consider is the extensional variant of the polymorphic HOL. We will call this logic $\mathrm{PHOL}_{\xi\eta}$, for reasons that will be discussed later (see Section 2.5).

$$
\begin{array}{rr}
\tau, \sigma ::= o & \text{(prop. formulas)} \\
\mid \iota & \text{(individuals)} \\
\mid \tau \Rightarrow \sigma & \text{(functions)} \\
\mid \alpha & \text{(type variables)} \\
\mid \Pi\alpha.\ \tau & \text{(polymorphic functions)}
\end{array}
$$

Figure 2.2: Type Syntax of Polymorphic Higher-Order Logic

$$
\frac{}{\mathfrak{R} \vdash o :_{\mathsf{R}} \mathsf{S}} \qquad \frac{}{\mathfrak{R} \vdash \iota :_{\mathsf{R}} \mathsf{S}}
$$

$$
\frac{\mathfrak{R} \vdash \tau_1 :_{\mathsf{R}} r_1 \qquad \mathfrak{R} \vdash \tau_2 :_{\mathsf{R}} r_2}{\mathfrak{R} \vdash \tau_1 \Rightarrow \tau_2 :_{\mathsf{R}} (r_1 \sqcup r_2)}
$$

$$
\frac{}{\mathfrak{R}, \alpha :_{\mathsf{R}} r \vdash \alpha :_{\mathsf{R}} r} \qquad \frac{\mathfrak{R}, \alpha :_{\mathsf{R}} \mathsf{S} \vdash \tau :_{\mathsf{R}} r}{\mathfrak{R} \vdash (\Pi\alpha.\ \tau) :_{\mathsf{R}} \mathsf{L}}
$$

Figure 2.3: The Ranking Rules of Polymorphic HOL

2.4.1 Ranks and Types

2.4⟨2⟩ Polymorphic Higher-Order Logic adds two additional types to System F: the type o of propositional formulas, those terms which have a truth value, and the type ι of individuals, the type of basic objects. The grammar of these types is described in Figure 2.2.

2.4⟨3⟩ Simply extending higher-order logic with type polymorphism is widely-known to be inconsistent [Coquand, 1995; Geuvers, 2007; Hurkens, 1995]. The approach we will take to resolve this is similar to the approach taken by HOL2P. We divide types into two ranks: large (L) and small (S), with $\mathsf{S} \leq \mathsf{L}$. Every type is ascribed one of these ranks, with $\Pi\alpha.\ \tau$ always creating a type of rank L, but requiring an α of rank S. In addition, type substitution is now required to respect ranks. The rules of the ranking system is described in Figure 2.3.

2.4⟨4⟩ The cause of the inconsistency has to do with impredicative polymorphism—that is, the ability to instantiate a polymorphic type with another polymorphic type. As the type variable of any polymorphic function must be S, but all types containing polymorphism are L, such impredicative instantiations are forbidden in polymorphic HOL.

Types and Terms

2.4⟨5⟩ As with all higher-order logics, we extend the lambda calculus with additional primitive terms representing logical connectives. In extensional polymorphic higher-order logic, the term syntax is extended with the equality symbol $[=]_\tau$ and the indefinite description symbol $[\varepsilon]_\tau$, for each type τ. The term grammar is given in Figure 2.4.

$$
\begin{array}{rll}
s, t, u, v ::= & x_\tau & \text{(variables)} \\
| & \lambda x_\tau.\ s & \text{(lambda abstraction)} \\
| & s\ t & \text{(application)} \\
| & \Lambda\alpha.\ s & \text{(type abstraction)} \\
| & s\ [:\tau:] & \text{(type application)} \\
| & [=]_\tau & \text{(equality)} \\
| & [\varepsilon]_\tau & \text{(indefinite description)}
\end{array}
$$

Figure 2.4: Term Syntax of Polymorphic Higher-Order Logic

2.4⟨6⟩ The binary function $[=]_\tau$ represents object-logic equality, i.e. the equality provable inside the logic; it has the type $\tau \Rightarrow \tau \Rightarrow o$. Equality is usually written an an infix operator $s = t$, which we interpret as an abbreviation for $[=]\ s\ t$.

2.4⟨7⟩ The function $[\varepsilon]_\tau$ represents 'indefinite description', also known as Hilbert's epsilon operator, and has the type $(\tau \Rightarrow o) \Rightarrow \tau$. It is usually written as a binder $(\varepsilon x.\ p)$, which we interpret as an abbreviation for $[\varepsilon]_\tau\ (\lambda x.\ p)$.

2.4⟨8⟩ To give some understanding of what indefinite description is, $\varepsilon x_\tau.\ p$ represents *some* object of type τ that satisfies the predicate p. To skip ahead a little, it is completely characterised by the inference rule

$$\frac{}{\vdash p\ s \longrightarrow p\ (\varepsilon x.\ p\ x)}\ \text{select-ax}$$

which says that to show that $p\ (\varepsilon x.\ p\ x)$ holds, it is enough to show that there is *some* s for which p holds.

2.4⟨9⟩ As shown in Figure 2.5, just adding $[=]$ is enough to define the common logical connectives (i.e. $\wedge$, $\longrightarrow$, etc.). In addition, in polymorphic HOL, we can define a polymorphic-forall $\forall_{\mathsf{ty}}\alpha.\ p$, which expresses that the formula s holds for all (small) types. These have the usual introduction and elimination rules (Subsection A.4.1). Moreover, using the connectives, injective and surjective functions can be defined in the usual manner.

2.4.2 Proof Rules

2.4⟨10⟩ The proof rules of $\mathrm{PHOL}_{\xi\eta}$ (Figure 2.6) belong to thee main groups: the structural rules, the equality rules, and axioms.

2.4⟨11⟩ The *structural* rules clarify how the antecedents may be used in a proof. In interpreting these rules, recall that the set of antecedents (Γ) is a multiset; that is, order doesn't matter, but the amount of elements does. The rule *assm* says that if a formula is

$$
\begin{aligned}
\top &\triangleq (\lambda p.\ p) =_{o \Rightarrow o} (\lambda p.\ p) \\
p \wedge q &\triangleq (\lambda f.\ f\ p\ q) = (\lambda f.\ f\ \top\ \top) \\
p \longrightarrow q &\triangleq (p \wedge q) = p \\
[\forall] &\triangleq \lambda p.\ p =_{o \Rightarrow o} (\lambda x.\ \top) \\
\forall x.\, p &\triangleq [\forall]\ (\lambda x.\ p) && \text{(where } p \text{ may contain } x\text{)} \\
[\exists] &\triangleq \lambda p.\ \forall z_o.(\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o \\
\exists x.\, p &\triangleq [\exists]\ (\lambda x.\ p) && \text{(where } p \text{ may contain } x\text{)} \\
p \vee q &\triangleq \forall z_o.(p \longrightarrow z_o) \longrightarrow (q \longrightarrow z_o) \longrightarrow z_o \\
\bot &\triangleq \forall z_o.\, z_o \\
\neg p &\triangleq p \longrightarrow \bot \\
[\forall_{\mathsf{ty}}] &\triangleq \lambda z.\ z =_{\Pi \alpha.\ o} (\Lambda \alpha.\ \top) \\
\forall_{\mathsf{ty}} \alpha.\ p &\triangleq [\forall_{\mathsf{ty}}]\ (\Lambda \alpha.\ p) && \text{(where } p \text{ may contain } \alpha\text{)}
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{inj}\ f &\triangleq \forall x\ y.\ f\ x = f\ y \longrightarrow x = y \\
\mathrm{onto}\ f &\triangleq \forall y.\ \exists x.\ f\ x = y
\end{aligned}
$$

Figure 2.5: The Connectives defined in Polymorphic Higher-Order Logic

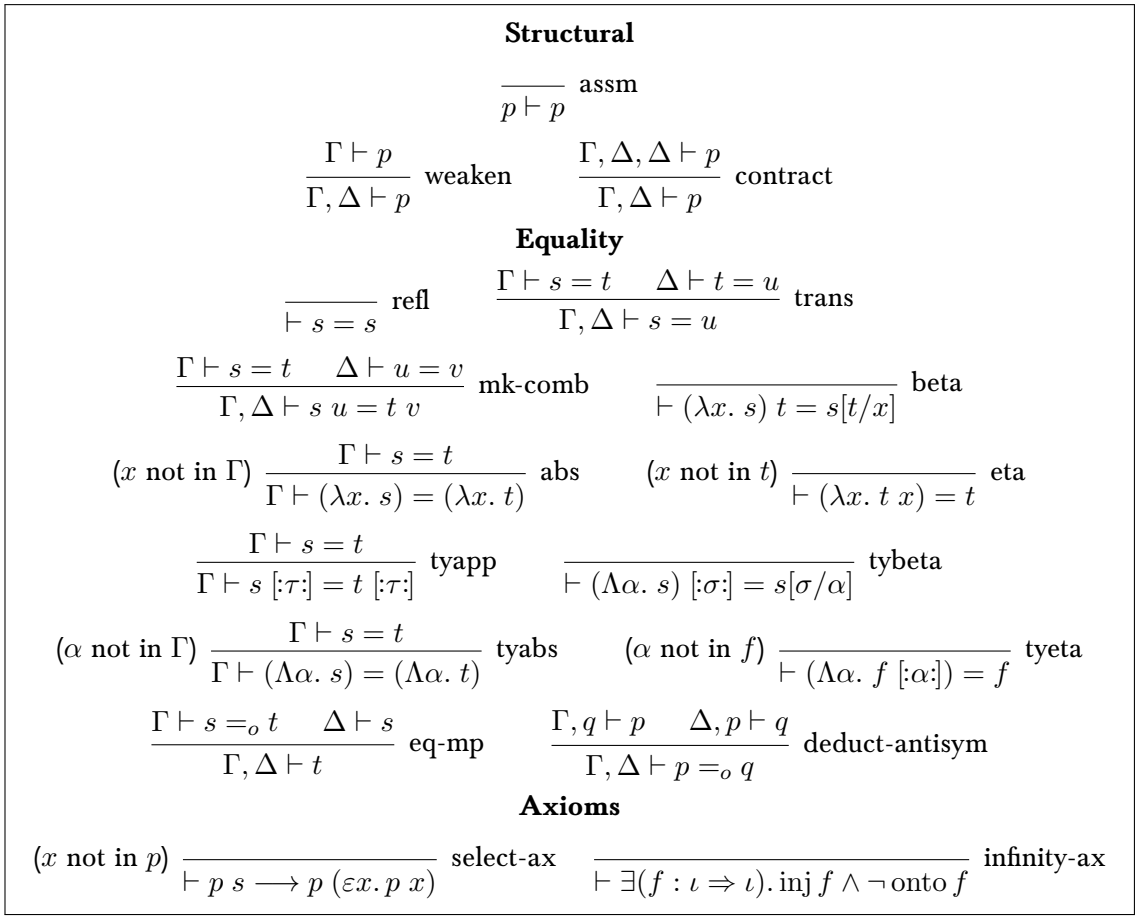


Figure 2.6: The proof rules of (Extensional) Polymorphic HOL

assumed, then it can be proven. The rule ***contract*** says that if a theorem can be proven with duplicated antecedents, then it can be proven with de-duplicated antecedents (this has the effect of making Γ act like a set). The rule ***weaken*** says that if it is possible to prove a theorem under some antecedents Γ, then the theorem is also provable under all multisets of antecedents that contain Γ.

2.4⟨12⟩ The rules for ***equality*** specify the behaviour of equality. ***refl*** says that equality is reflexive, and ***trans*** says that it's transitive. ***mk-comb*** says that application is a congruence relation, that is, to show that $s\ u$ is equal to $t\ v$, is is sufficient to show that s is equal to t, and u to v. The rule ***abs*** says that to show to lambda functions are equal, it is enough to show that their bodies are equal. ***eta*** says that abstracting and immediately applying the abstracted variable to a function f (so long as that variable does not occur in f), is exactly the same as f. The rules ***tyapp***, ***tyabs***, and ***tyeta*** are the type-abstraction equivalents of ***mk-comb***, ***abs***, and ***eta***.

2.4⟨13⟩ The rules ***eq-mp*** and ***deduct-antisym*** define equality on formulas. ***eq-mp*** says that if two formulas p and q are equal, then a proof of p is also a proof of q. ***deduct-antisym*** essentially says that if you can show p by assuming q, and q by assuming p, then they are equal.

2.4⟨14⟩ Lastly, the ***axioms*** are rules which define the character of the logic. The rule ***select-ax*** defines the ε operator to be the indefinite description operator, and the rule ***infinity-ax*** enforces that there are a countably infinite number of members of the type ι.

2.4.3 Derived and Admissible Rules

2.4⟨15⟩ With any logical system, there are two ways of demonstrating a proof rule holds (note that theorems of a logic are just proof rules with no assumptions). The first is showing that the proof rule is ***derivable***, that is, reasoning from the assumptions, the conclusion can be reached by application of axiomatic or (other) derivable rules.

2.4⟨16⟩ The second is by showing that a rule is ***admissible***, that is, that a proof-tree can always be constructed which demonstrates the rule. This is a weaker notion than derivability, as adding additional proof-rules or ways to construct formulae can invalidate the construction. Nevertheless, showing that a rule is admissible still gives license to use it as a derivation step in proof construction.

2.4⟨17⟩ The following rules are derivable or admissible in $\mathrm{PHOL}_{\xi\eta}$

$$\frac{\Gamma \vdash p}{\Gamma[\theta] \vdash p[\theta]}\ \text{inst} \qquad \frac{\Gamma \vdash p}{\Gamma[\phi] \vdash p[\phi]}\ \text{ty-inst}$$

$$\frac{\Gamma \vdash s = t \quad \Delta \vdash e[s]}{\Gamma, \Delta \vdash e[t]}\ \text{rewrite} \qquad \frac{\Gamma \vdash s \quad s, \Delta \vdash t}{\Gamma, \Delta \vdash t}\ \text{ante-subst}$$

Rewriting

2.4⟨18⟩ If two terms differ only in the structure of some identically-placed subterm, and those subterms are provably equal, it is possible to rewrite the subterm to show the equality of the larger terms. This rule is called *rewrite*

$$\frac{\Gamma \vdash s = t \qquad \Delta \vdash e[s]}{\Gamma, \Delta \vdash e[t]} \text{ rewrite}$$

where $e[\cdot]$ represents the common structure surrounding the subterms. The proper formalisation of this rule relies on the notion of a context (that is, a term with a hole, which stands for an unsubstituted subterm, which *may* capture bound variables). A proof of the admissibility of this result is given in Section A.2.

2.4.4 Antecedent Substitution

2.4⟨19⟩ The rule of antecedent-substitution says that if you wish to prove $\Gamma, \Delta \vdash t$, it is sufficient to show $s, \Delta \vdash t$ under some antecedent s, and that s is provable from Γ.

$$\frac{\Gamma \vdash s \qquad s, \Delta \vdash t}{\Gamma, \Delta \vdash t} \text{ ante-subst}$$

Proof. The proof is as follows[1]

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash s}{\Gamma, t \vdash s}\text{ weaken} \qquad s, \Delta \vdash t}{\Gamma, \Delta \vdash s =_o t}\text{ deduct-antisym} \qquad \Gamma \vdash s}{\dfrac{\Gamma, \Delta, \Gamma \vdash t}{\Gamma, \Delta \vdash t}\text{ contract}}\text{ eq-mp}$$

□

2.4⟨20⟩ Note that this is not *cut* of the sequent calculus, as this logic is a natural deduction system. The key difference is that reasoning backwards in a sequent calculus with the subformula property cannot introduce any term which does not occur in the conseqents and antecendents already present; this means that a proof of cut-admissibility *must* be done by induction over the proof. The natural deduction calculus has no such restriction (for backwards reasoning, at least) as is evidenced by elimination rules such as *eq-mp* in which the judgement $\Gamma \vdash s =_o t$ may contain a connective ($=_o$) not present in either s or t.

[1] My thanks to Johannes Åman Pohjola, who pointed out this simple proof using just the primitive PHOL rules. Note that a similar derivation can be performed using ∨I and ∨E.

Instantiation

2.4⟨21⟩ The instantiation rules ***inst*** and ***tyinst*** state that a general proof of some fact p under the antecedents Γ also is a proof of any more specific instantiation of the variables or type-variables of that theorem. These rules are admissible to polymorphic higher-order logic; the proof of this result is a straightforward but mechanical induction over the proof of $\Gamma \vdash p$, applying the instantiation the the hypotheses that produced the $\Gamma \vdash p$. A full proof of this can be found in Section A.5. They are stated

$$\frac{\Gamma \vdash p}{\Gamma[\theta] \vdash p[\theta]}\ \text{inst} \qquad \frac{\Gamma \vdash p}{\Gamma[\phi] \vdash p[\phi]}\ \text{ty-inst}$$

2.5 Function Extensionality

2.5⟨1⟩ ***Functional extensionality*** is a property of equality that states that two functions are equal when on equal inputs they produce equal outputs. Formally this rule is written

$$(x \text{ not free in } \Gamma, f, g)\ \frac{\Gamma \vdash f\ x = g\ x}{\Gamma \vdash f = g}\ \text{fun-ext}$$

2.5⟨2⟩ Extensional polymorphic higher-order logic has functional extensionality (as the name implies), as the above rule can be derived from ***abs*** and ***eta*** as follows

$$\dfrac{(x \text{ not free in } f)\ \dfrac{}{\vdash (\lambda x.\ f\ x) = f}\ \text{eta} \quad (x \text{ not free in } g)\ \dfrac{}{\vdash (\lambda x.\ g\ x) = g}\ \text{eta} \quad (x \text{ not free in } \Gamma)\ \dfrac{\Gamma \vdash f\ x = g\ x}{\Gamma \vdash (\lambda x.\ f\ x) = (\lambda x.\ g\ x)}\ \text{abs}}{\Gamma \vdash f = g}\ \text{rewrite}^+$$

2.5⟨3⟩ In the theory of the lambda calculus, ***abs*** is also known as weak extensionality (ξ). It is a widely-known result that weak extensionality (ξ) and eta-equivalence (η) are equivalent to function extensionality [Barendregt, 1981, p.32].

2.6 Intensional Polymorphic Higher-Order Logic

2.6⟨1⟩ The intensional polymorphic HOL will be defined by removing the rules ***abs*** and ***eta*** (and their polymorphic-function equivalents ***tyabs*** and ***tyeta***). This requires a number of changes to the calculus.

2.6⟨2⟩ Firstly, note that without ***abs***, the applicability of the ***rewrite*** rule is restricted. The rewrite rule can no longer be applied to rewrite any subterm which contains a bound variable (this is related to the notion of weak λ-reduction in [Çaman and Hindley, 1998]). Indeed, the rewrite lemma now becomes

$$(x \text{ is free in } u)\ \frac{\Gamma \vdash s = t \qquad \Delta \vdash u[s/x]}{\Gamma, \Delta \vdash u[t/x]}\ \text{rewrite}$$

using standard capture-avoiding substitution. The admissibility of the rules ***inst***, ***ty-inst*** are not affected by the intensionality.

2.6⟨3⟩ Moreover, the removal of ***abs*** and ***tyabs*** results in a problem with the definition of the universal quantifier, as the introduction rules for $[\forall]_\tau$ and $[\forall_{\mathsf{ty}}]$ use ***abs*** and ***tyabs*** respectively (Section A.4.1). To address this, we add $[\forall]_\tau$ and $[\forall_{\mathsf{ty}}]$ as primitive symbols to the intensional logic.

$$\begin{aligned} s, t, u, v ::= {} & \ldots \\ & \mid [\forall]_\tau && \text{(forall)} \\ & \mid [\forall_{\mathsf{ty}}] && \text{(type-forall)} \end{aligned}$$

These symbols are associated with the additional rules

$$(x \text{ not in } \Gamma)\ \frac{\Gamma \vdash p}{\Gamma \vdash \forall x.\, p}\ \forall\text{I} \qquad \frac{\Gamma \vdash [\forall]\ p}{\Gamma \vdash p\ s}\ \forall\text{E}$$

$$(\alpha \text{ not in } \Gamma)\ \frac{\Gamma \vdash p}{\Gamma \vdash \forall_{\mathsf{ty}} \alpha.\ p}\ \forall_{\mathsf{ty}}\text{I} \qquad \frac{\Gamma \vdash [\forall_{\mathsf{ty}}]\ p}{\Gamma \vdash p\ [\!:\tau:\!]}\ \forall_{\mathsf{ty}}\text{E}$$

2.6⟨4⟩ Note that proving that this system actually is intensional is left as future work; reasons why it might not be are if ***abs*** and ***eta*** could be derived from $[\forall]$, $[\forall_{\mathsf{ty}}]$, $[\varepsilon]$, or $[\dot{=}]_\alpha$. One means of demonstrating such a result is to construct an intensional Henkin semantics [Henkin, 1950] for the logic.

Chapter 3

Previous Work

3.0⟨1⟩ In this chapter, we review previous work on extensionality translations, with a particular focus on higher-order logics.

3.1 The Removal of Functional Extensionality by Translation

3.1⟨1⟩ An interesting fact is that any HOL (without polymorphism or type variables) proof that uses functional extensionality can be embedded into HOL without a functional extensionality rule. This is proven by means of a logical translation between the theorems of the extensional and intensional higher-order logics.

3.1⟨2⟩ As this result is generally useful, a number of variants has been produced [Takeuti, 1953; Gandy, 1956; Luckhardt, 1973; Kohlenbach, 2008; Rizkallah, 2009]. The first translation to demonstrate this results for higher-order logic was Gandy's translation [Gandy, 1956], which uses two mutually recursive relations to define an extensional partial-equivalence and a predicate which verifies whether a term may be treated extensionally. A simpler translation was presented by Rizkallah [2009], which uses a single recursion to define a partial equivalence relation.

3.1⟨3⟩ The crux of all these translations is a partial-equivalence acts like extensional equality under certain assumptions.

Gandy's Translation

3.1⟨4⟩ The first translation is one by Gandy [1956], a translation from extensional to intensional versions of Church's Simple Theory of Types. The aim of this work is to reduce a consistency proof of extensional HOL to that of intensional HOL, in that a proof of

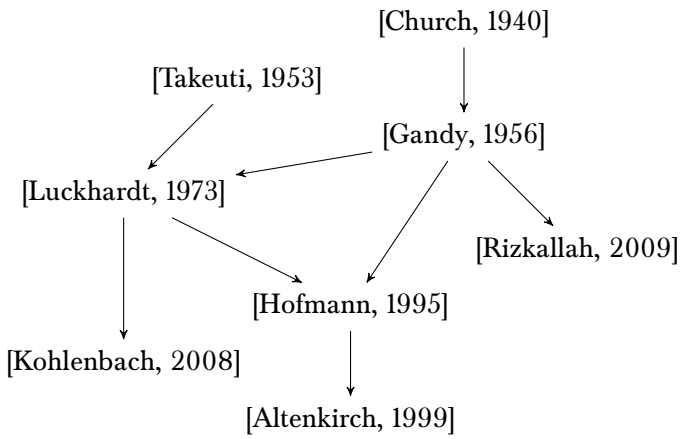


Figure 3.1: A sketch of the relations between the ideas in each paper

consistency of intensional HOL, along with Gandy's translation, immediately allows one to show that extensional HOL is consistent.

3.1⟨5⟩ To see why consistency follows, imagine we had a proof of False in extensional higher order logic, and that intensional HOL was consistent. Then, by the translation, we could obtain some translated proof of False (conditional on the fact that False translates to False) in the intensional HOL, violating consistency.

3.1⟨6⟩ Gandy's version of HOL is based on Church's simple theory of types (the original HOL), which is a classical higher-order logic defined by a Hilbert-system. The significant differences from $\mathrm{PHOL}_{\xi\eta}$ are as follows: it has only the types o, ι, and $\tau_1 \Rightarrow \tau_2$, and contains a primitive forall $[\forall]$, disjunction $[\vee]$, and negation $[\neg]$; the rest of the logical connectives are defined in terms of these.

3.1⟨7⟩ Intensional equality is introduced by defining Leibniz equality (that is, objects which agree on all predicates are equal) with the following definition

$$[=]_\tau \triangleq \lambda s_\tau\ t_\tau.\ \forall p.\, p\ s \longrightarrow p\ t.$$

Extensionality is introduced into the logic by defining the relation $[\overset{\mathrm{E}}{=}]_\tau$ by induction on τ as

$$\begin{aligned} [\overset{\mathrm{E}}{=}]_o &\triangleq \lambda p\ q.\ p \longleftrightarrow q \\ [\overset{\mathrm{E}}{=}]_{\tau\Rightarrow\sigma} &\triangleq \lambda f\ g.\ \forall x_\tau.\, f\ x \overset{\mathrm{E}}{=}_\sigma g\ x \\ [\overset{\mathrm{E}}{=}]_\iota & \qquad\qquad \text{(primitive symbol)} \end{aligned}$$

where $[\overset{\mathrm{E}}{=}]_\iota$ is axiomatised to be an equivalence relation. (Note that this means that $[\overset{\mathrm{E}}{=}]_\iota$ can be coarser than Lebiniz equality.) Finally, the logic is made extensional by adding the axiom $\forall x\ y.\, x \overset{\mathrm{E}}{=} y \longrightarrow x = y$.

3.1⟨8⟩ The main project of Gandy's translation is then to define a translation $(-)^{\mathbf{F}}$ such

that

$$\vdash (\forall x\ y.\, x \overset{\mathrm{E}}{=} y \longrightarrow x = y)^{\mathbf{F}}$$

is provable in the intensional system. To do this, Gandy defines a relation $[\overset{\mathrm{G}}{=}]_\tau$ and a predicate Mod_τ, by mutual induction over the structure of types.

$$x \underset{o}{\overset{\mathrm{G}}{=}} y \triangleq x \longleftrightarrow y \qquad \mathrm{Mod}_o\, x \triangleq x \longleftrightarrow x$$

$$x \underset{\iota}{\overset{\mathrm{G}}{=}} y \triangleq x \underset{\iota}{\overset{\mathrm{E}}{=}} y \qquad \mathrm{Mod}_\iota\, x \triangleq x \underset{\iota}{\overset{\mathrm{E}}{=}} x$$

$$f \underset{\tau\Rightarrow\sigma}{\overset{\mathrm{G}}{=}} g \triangleq \forall x_\tau.\, \mathrm{Mod}\ x \to f\, x \underset{\sigma}{\overset{\mathrm{G}}{=}} g\, x$$

$$\mathrm{Mod}_{\tau\Rightarrow\sigma}\, f \triangleq \forall x\ y.\, \mathrm{Mod}\ x \wedge \mathrm{Mod}\ y \wedge x \overset{\mathrm{G}}{=} y \longrightarrow \mathrm{Mod}(f\, x) \wedge f\, x \overset{\mathrm{G}}{=} f\, y$$

3.1⟨9⟩ To give the intuition behind this construction, Mod is an ***extensional domain*** predicate which is provable exactly when we are allowed to treat an object extensionally or not; i.e. whether it's in the domain of the extensional partial-equivalence.

3.1⟨10⟩ Gandy defined the translation of a term as follows

$$\begin{aligned}
x^{\mathbf{G}} &\triangleq x && \text{(where } x \text{ is a var)}\\
(\lambda x.\ s)^{\mathbf{G}} &\triangleq \lambda x.\ s^{\mathbf{G}}\\
(s\, t)^{\mathbf{G}} &\triangleq s^{\mathbf{G}}\, t^{\mathbf{G}}\\
[\longrightarrow]^{\mathbf{G}} &\triangleq [\longrightarrow]\\
[\forall]^{\mathbf{G}} &\triangleq \lambda f.\ \forall x.\, \mathrm{Mod}\ x \longrightarrow f^{\mathbf{G}}\, x\\
[\overset{\mathrm{E}}{=}]^{\mathbf{G}} &\triangleq [\overset{\mathrm{G}}{=}]
\end{aligned}$$

Note how $[\forall]$ has a Mod inserted as a premise that needs to be proven. This restricts the objects the forall can be instantiated with to those which behave extensionally.

3.1⟨11⟩ Finally, Gandy's translation says that if

$$\Gamma \underset{\text{G-E}}{\vdash} s$$

holds in his extensional HOL, then

$$\mathrm{Mod}\, x_1, \mathrm{Mod}\, x_2, \ldots, \mathrm{Mod}\, x_n, \Gamma^{\mathbf{G}} \underset{\text{G-I}}{\vdash} s^{\mathbf{G}}$$

$$\text{(where } \mathrm{FV}(s) \cup \mathrm{FV}(\Gamma) = \{x_1, x_2, \ldots, x_n\}\text{)}$$

in his intensional HOL. This implies that

$$\vdash (\forall x\ y.\, x \overset{\mathrm{E}}{=} y \longrightarrow x = y)^{\mathbf{F}}$$

holds, as $\forall x\ y.\, x \overset{\mathrm{E}}{=} y \longrightarrow x = y$ has no free variables.

3.1⟨12⟩ Gandy's translation is one of the first translations of extensional HOL. The mutually recursive definition of his extensional partial-equivalence is complicated, but allows a separation between the concepts of being an extensional value, and being extensionally equal.

Rizkallah's Translation

3.1⟨13⟩ Rizkallah [2009; Brown and Rizkallah, 2013] presents a translation from an extensional, classical HOL to an intensional intuitionistic HOL. The proofs rules of the former are specified using a tableau calculus, and the latter a natural deduction system. The translation is achieved by combining a double-negation style transformation (a modification of Kuroda's classical to intuitionistic translation in first-order logic [Kuroda, 1951]) and an extensionality translation.

3.1⟨14⟩ The main idea of the extensionality translation is, similar to Gandy [1956], to define an extensional partial-equivalence which simulates extensional equality in the intensional logic. The extensional partial-equivalence is defined as follows

$$\begin{aligned} x \overset{\mathrm{R}}{\underset{o}{=}} y &\triangleq x \longleftrightarrow y \\ x \overset{\mathrm{R}}{\underset{\iota}{=}} y &\triangleq \forall p.\, p\ x \longrightarrow p\ y \\ f \overset{\mathrm{R}}{\underset{\tau\Rightarrow\sigma}{=}} g &\triangleq \forall x\ y.\, x \overset{\mathrm{R}}{\underset{\tau}{=}} y \longrightarrow \neg\neg(f\ x \overset{\mathrm{R}}{\underset{\sigma}{=}} g\ y) \end{aligned}$$

This definition removes the mutually recursive extensional-domain predicate found in Gandy's translation. Instead, in Rizkallah's translation, the formula $x \overset{\mathrm{R}}{=} x$ functions as the domain predicate.

3.1⟨15⟩ Terms are mapped in a similar manner to Gandy's translation, however, double negations are added on the forall due to the fact the translation is also mapping into an intuitionistic HOL.

$$\begin{aligned} x^{\mathbf{R}} &\triangleq x && \text{(where } x \text{ is a var)} \\ (\lambda x.\ s)^{\mathbf{R}} &\triangleq \lambda x.\ s^{\mathbf{G}} \\ (s\,t)^{\mathbf{R}} &\triangleq s^{\mathbf{G}}\,t^{\mathbf{G}} \\ [\longrightarrow]^{\mathbf{R}} &\triangleq [\longrightarrow] \\ [\forall]^{\mathbf{R}} &\triangleq \lambda f.\ \forall x.\, x \overset{\mathrm{R}}{=} x \longrightarrow \neg\neg(f^{\mathbf{R}}\,x) \\ [=]^{\mathbf{R}} &\triangleq [\overset{\mathrm{R}}{=}] \end{aligned}$$

3.1⟨16⟩ Theorems are mapped as follows: if

$$\Gamma \underset{\text{R-E}}{\vdash} s$$

holds in extensional, classical HOL, then

$$x_1 = x_1, x_2 = x_2, \ldots, x_n = x_n, \Gamma^{\mathbf{R}}, \neg s^{\mathbf{R}} \underset{\text{R-I}}{\vdash} \bot$$

$$\text{(where } \mathrm{FV}(s) \cup \mathrm{FV}(\Gamma) = \{x_1, x_2, \ldots, x_n\}\text{)}$$

holds in intensional, intuitionistic HOL.

3.1⟨17⟩ Focusing on the extensionality transformation, this translation represents a useful simplification over Gandy's method, removing the mutual recursion. Rizkallah in particular

points out that the translation of extensional equality into the intensional extensional partial-equivalence is a partial equivalence relation, for which reflexivity on terms holds only when every variable in that term can be proven to be reflexive.

3.1.1 Extensionality Translations in Type Theory

3.1⟨18⟩ Extensionality translations are also a well studied area in *type theory* [Altenkirch, 1999; Hofmann, 1995; Barthe et al., 2003]. In this area, ways to remove extensionality from a type theory are valuable as in extensional type theory, equality and type-checking are undecidable, and introducing an axiomatic extensionality to intensional type theory allows the construction of terms do not reduce to constructors of their type.

3.1⟨19⟩ These translations are formulated in terms of *setoids* and *partial setoids*, which perform the function of the partial-equivalences used in Gandy's and Rizkallah's translations. A setoid is a collection of objects, called the *carrier*, along with an equivalence relation on the elements of that set, called the *book equality* for that setoid. A partial setoid is like a setoid except it has a *partial* equivalence relation, that is, an equivalence relation without reflexivity. A structure preserving map from one (partial) setoid to another is a translation of equality.

Chapter 4

Translation

4.0⟨1⟩ In this chapter we develop a translation from $\mathrm{PHOL}_{\xi\eta}$ to PHOL. This translation removes functional extensionality from the logic, and demonstrates an embedding of proofs making use of extensionality into a logic without it.

4.1 Definition of the Translation

4.1.1 The Translation of Terms

4.1⟨1⟩ The translation of terms is conceptually very simple, but contains several tricky components. On terms of $\mathrm{PHOL}_{\xi\eta}$, the map $(-)^\bullet$ acts in the following manner

$$\begin{array}{lcl} x_\tau^\bullet & \triangleq & x_\tau \\ (\lambda x_\tau.\ s)^\bullet & \triangleq & \lambda x_\tau.\ s^\bullet \\ (s\ t)^\bullet & \triangleq & s^\bullet\ t^\bullet \\ (\Lambda\alpha.\ s)^\bullet & \triangleq & \Lambda\alpha.\ s^\bullet \\ (s\ [:\tau:])^\bullet & \triangleq & s^\bullet\ [:\tau:] \\ ([=]_\tau)^\bullet & \triangleq & [\dot{=}]_\tau \end{array}$$

this simply replaces every equality of $\mathrm{PHOL}_{\xi\eta}$ with the symbol $[\dot{=}]_\tau$.

4.1⟨2⟩ The symbol $[\dot{=}]_\tau$ represents an extensional partial-equivalence defined in PHOL; that is, a binary relation on terms $[\dot{=}]_\tau : \tau \Rightarrow \tau \Rightarrow o$ that is transitive and symmetric, but not necessarily reflexive. This extensional partial equivalence acts like equality with functional extensionality under certain assumptions, but is defined in such a way that it does not add

any new theorems or terms to the logic.

4.1⟨3⟩ The $[\dot{=}]_\tau$ term is defined by recursion on the type τ of its argument, as follows

$$
\begin{array}{lcll}
[\dot{=}]_o & \triangleq & [=]_o & \\
[\dot{=}]_\iota & \triangleq & [=]_\iota & \\
[\dot{=}]_{\tau_1 \Rightarrow \tau_2} & \triangleq & \lambda f\ g.\ \forall x\ y.\ x \mathrel{\dot{=}}_{\tau_1} y \longrightarrow f\ x \mathrel{\dot{=}}_{\tau_2} g\ y & \text{(for } x \text{ and } y \text{ not free in } f \text{ and } g\text{)} \\
[\dot{=}]_{\Pi\alpha.\ \tau} & \triangleq & \lambda f\ g.\ \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow f\ [{:}\alpha{:}] \mathrel{\dot{=}}_\tau g\ [{:}\alpha{:}] & \text{(for } \alpha \text{ not free in } f \text{ and } g\text{)} \\
[\dot{=}]_\alpha & & & \text{(primitive symbol)}
\end{array}
$$

4.1⟨4⟩ For this to work, we must amend the terms of PHOL with a new symbol $[\dot{=}]_\alpha$, defined for each type variable α. The reason for this addition is that our translation must respect type-substitution—that is $s^\bullet[\tau/\alpha]$ must equal $(s[\tau/\alpha])^\bullet$, but nothing in the logic allows us to express this. To ensure the safety of this addition, this symbol is ***undefined***, in that no proof rule refers to it explicitly. Note that this constitutes a significant difference to earlier translations—such as Rizkallah's [Rizkallah, 2009]—where $[\dot{=}]$ never occurs in any concrete translated theorem.

4.1⟨5⟩ To ensure the translation respects type-substitution, the symbol $[\dot{=}]_\alpha$ has an interaction with type-substitution. When a type is substituted for α, $[\dot{=}]_\alpha$ reduces in the following manner

$$
\begin{gathered}
[\dot{=}]_\alpha[o/\alpha] \triangleq [\dot{=}]_o \\
[\dot{=}]_\alpha[\iota/\alpha] \triangleq [\dot{=}]_\iota \\
[\dot{=}]_\alpha[\tau_1 \Rightarrow \tau_2/\alpha] \triangleq [\dot{=}]_{\tau_1 \Rightarrow \tau_2} \\
[\dot{=}]_\alpha[\Pi\alpha.\ \tau/\alpha] \triangleq [\dot{=}]_{\Pi\alpha.\ \tau} \\
[\dot{=}]_\alpha[\beta/\alpha] \triangleq [\dot{=}]_\beta
\end{gathered}
$$

4.1⟨6⟩ On booleans (o) and individuals (ι), $[\dot{=}]$ is simply the basic equality of HOL without functional extensionality.

4.1⟨7⟩ For functions ($\tau_1 \Rightarrow \tau_2$), two functions f and g are considered equivalent by $[\dot{=}]$ whenever, given two inputs which are extensionally partially-equivalent, the results of their application to f and g are extensionally partially-equivalent. This is essentially the principle of functional extensionality, but stated explicitly anywhere it could be used.

4.1⟨8⟩ On the polymorphic type ($\Pi\alpha.\ \tau_1$), the left and right terms, s and t, are considered to be extensionally equivalent when they are extensionally equivalent for all type variables α for which $[\dot{=}]_\alpha$ is reflexive, transitive, symmetric, and where $[\dot{=}]_\alpha$ is extensionally equivalent to itself.

4.1⟨9⟩ These requirements are collected into the formula $\mathcal{E}(\alpha)$, defined as the conjunction

of the equations

$$\begin{aligned}
&\left(\forall x\ y.\ x \mathrel{\dot{=}}_\alpha y \longrightarrow y \mathrel{\dot{=}}_\alpha x\right) && \text{(symmetric)}\\
&\left(\forall x\ y\ z.\ x \mathrel{\dot{=}}_\alpha y \longrightarrow y \mathrel{\dot{=}}_\alpha z \longrightarrow x \mathrel{\dot{=}}_\alpha z\right) && \text{(transitive)}\\
&\left(\exists x.\ x \mathrel{\dot{=}}_\alpha x\right) && \text{(existence of a reflexive term)}\\
&\left([\dot{=}]_\alpha \mathrel{\dot{=}}_{\alpha\Rightarrow\alpha\Rightarrow o} [\dot{=}]_\alpha\right) && \text{(extensionally reflexive)}
\end{aligned}$$

4.1⟨10⟩ Furthermore, we overload $\mathcal{E}$ to types and terms. On types, $\mathcal{E}(\tau)$ is simply the multiset of free type-variables in τ. On terms, it is simply

$$\mathcal{E}(s) \triangleq \{\!\{\mathcal{E}(\alpha) \mid \alpha \in \mathrm{FV}_{\mathrm{ty}}(s)\}\!\}$$

4.1⟨11⟩ To make this definition work for in the final translation, we also need another set of formulas $\mathcal{R}$, that is the collection of all reflexive equivalences on the free variables of a term.

$$\mathcal{R}(s) \triangleq \{\!\{x \mathrel{\dot{=}}_\tau x \mid x_\tau \in \mathrm{FV}(s)\}\!\}$$

4.1⟨12⟩ The final translation takes a theorem of $\mathrm{PHOL}_{\xi\eta}$

$$\Gamma \vdash_{\mathrm{PHOL}_{\xi\eta}} s$$

to the theorem of PHOL

$$\mathcal{E}(\Gamma^\bullet), \mathcal{E}(s^\bullet), \mathcal{R}(\Gamma^\bullet), \mathcal{R}(s^\bullet), \Gamma^\bullet \vdash_{\mathrm{PHOL}} s^\bullet$$

4.1⟨13⟩ A useful specialisation of this to note is when s is $t_1 =_\tau t_2$. Note that $\mathcal{E}([\dot{=}]_\tau) \equiv \mathcal{E}(\tau)$, as the only free $[\dot{=}]_\alpha$ in $[\dot{=}]_\tau$ are the free type variables in τ, and $\mathcal{R}([\dot{=}]_o) \equiv \emptyset$, as there are no free variables in $[\dot{=}]_\tau$.

$$\mathcal{E}(\Gamma^\bullet), \mathcal{E}(\tau), \mathcal{E}(t_1^\bullet), \mathcal{E}(t_2^\bullet), \mathcal{R}(\Gamma^\bullet), \mathcal{R}(t_1^\bullet), \mathcal{R}(t_2^\bullet), \Gamma^\bullet \vdash_{\mathrm{PHOL}} t_1^\bullet \mathrel{\dot{=}}_\tau t_2^\bullet$$

4.2 Translation Lemmas

4.2.1 Respects Substitution

4.2⟨1⟩ An important property of the translation of terms is that it respects substitution. This implies that a substitution inside a translation is equivalent to a substitution outside a translation, which allows for the interchange of substitution and translation. These lemmas are important for proving the correct translation of ***inst***, ***tyinst***, beta-equivalence, and generally anywhere where a substitution occurs under a $\bullet$.

4.2⟨2⟩ The following theorems can be shown by induction on s: for term substation, $(s[t/x])^\bullet$ is the same as $s^\bullet[t^\bullet/x]$ (***subst•***, B.2.1); and for type substitution, $(s[\tau/\alpha])^\bullet$ is the same as $s^\bullet[\tau/\alpha]$. (***tysubst•***, B.2.2)

4.2.2 Respects Rewriting

4.2⟨3⟩ As the translation respects substitution it also respects rewriting (B.2.3); that is

$$\frac{\Gamma \vdash s = t \qquad \Delta \vdash (e[s])^\bullet}{\Gamma, \Delta \vdash (e[t])^\bullet}\ \text{rewrite}$$

(here we overload the rule-name *rewrite* for convenience) The validity of this can be proven by noting that $e^\bullet[t^\bullet]$ is $(e[t])^\bullet$ by ***subst•***. This rule justifies, among other things, the use of beta underneath translation, as $((\lambda x.\ s)\ t)^\bullet =_\tau s^\bullet[t^\bullet/x]$.

4.2.3 Dropping Unused Translation Antecedents

4.2⟨4⟩ When showing that the translated rules are provable, some rules—for example, ***trans***, ***mk-comb***, ***eq-mp***—pose a problem, as their hypotheses may contain more variables than their conclusion.

4.2⟨5⟩ This causes a problem as the translation adds the additional antecedents $\mathcal{R}$ and $\mathcal{E}$ which use the variables in a theorem. In order to correctly obtain the conclusion, these additional antecedents must be eliminated in some manner. If this step is not performed, each translated theorem would become dependent on the manner by which it was proven. Note that this cannot be solved by an application of weakening, as weakening adds extra antecedents to the conclusion, not the hypotheses.

4.2⟨6⟩ The solution is to instantiate these leftover antecedents with more concrete terms and types for which $\mathcal{R}$ and $\mathcal{E}$ are true. As the variables don't appear in the conclusion, such an instantiation does not affect it. This is why $\mathcal{R}$ and $\mathcal{E}$ are assumed for the antecedents as well as the consequent of each theorem. Without these additional antecedents, some antecedents would be changed by this instantiation step.

4.2⟨7⟩ These proofs will require finding terms for $\mathcal{R}$ which are always reflexive—we call these terms Z_τ—and finding a type for which $\mathcal{E}$—this is simply o.

Reflexive Terms

4.2⟨8⟩ Terms for which $\mathcal{R}$ is true can be constructed as follows. Define the term Z_τ by recursion on its type

$$\begin{aligned}
\mathrm{Z}_o &\triangleq \top \\
\mathrm{Z}_\iota &\triangleq z && \text{(for some } z \text{ in } \iota) \\
\mathrm{Z}_{\tau \Rightarrow \sigma} &\triangleq \lambda x.\ \mathrm{Z}_\sigma && \text{(where } x \notin \mathrm{FV}(Z_\sigma)) \\
\mathrm{Z}_{\Pi\alpha.\ \tau} &\triangleq \Lambda\alpha.\ \mathrm{Z}_\tau \\
\mathrm{Z}_\alpha &\triangleq \varepsilon z.\ z \stackrel{\bullet}{=}_\alpha z
\end{aligned}$$

Note that this construction depends on the fact that the ι type is non-empty, which allows us to pick a z for ι. Moreover, the α case uses indefinite description to specify "the term z which satisfies $z \stackrel{\bullet}{=}_\alpha z$". This works because the $\mathcal{E}$ antecedents for α include $\exists z.\, z \stackrel{\bullet}{=}_\alpha z$, which allows the construction of such a term.

4.2⟨9⟩ Under the assumption of $\mathcal{E}$ for the type of Z, this term can be proven reflexive; that is

$$\frac{}{\mathcal{E}(\tau) \vdash \mathrm{Z}_\tau \stackrel{\bullet}{=}_\tau \mathrm{Z}_\tau}\ \text{reflZ}$$

The proof is by induction on τ; the full proof can be found in Subsection B.2.5.

drop-unused$\mathcal{R}$

4.2⟨10⟩ The rule *drop-unused*$\mathcal{R}$ (B.2.6) is

$$(\overline{x} \cap (\mathrm{FV}(\Gamma) \cup \mathrm{FV}(p)) = \emptyset)\ \frac{\mathcal{R}(\overline{x}), \Gamma \vdash p}{\mathcal{E}(\overline{x}), \Gamma \vdash p}\ \text{drop-unused}\mathcal{R}$$

This rule states that $\mathcal{R}$ antecedents on variables which do not occur anywhere else in the theorem may be reduced to $\mathcal{E}$ on those variables.

4.2⟨11⟩ The proof proceeds by constructing a substitution θ_Z, which substitutes Z_τ for every x_τ in $\overline{x}$. By inst, we can show

$$\mathcal{R}(\overline{x})[\theta_\mathrm{Z}], \Gamma[\theta_\mathrm{Z}] \vdash p[\theta_\mathrm{Z}]$$

and, as no variable of $\overline{x}$ occurs in Γ or p, and by unfolding the definition of $\mathcal{R}$, this theorem simplifies to

$$\{\!\{\mathrm{Z}_\tau \stackrel{\bullet}{=} \mathrm{Z}_\tau \mid x_\tau \in \overline{x}\}\!\}, \Gamma \vdash p.$$

4.2⟨12⟩ Using the theorem *reflZ*, under the antecedents $\mathcal{E}(\overline{x})$, every member of

$$\{\!\{\mathrm{Z}_\tau \stackrel{\bullet}{=} \mathrm{Z}_\tau \mid x_\tau \in \overline{x}\}\!\}$$

is true, and thus all these antecedents can be removed by repeated application of *ante-subst.* This completes the proof of *drop-unused*$\mathcal{R}$

drop-unused$\mathcal{E}$

4.2⟨13⟩ Similarly to *drop-unused*$\mathcal{R}$, we also require a theorem to drop $\mathcal{E}$ antecedents. The rule *drop-unused*$\mathcal{E}$ (B.2.8) is

$$(\overline{\alpha} \cap (\mathrm{FV}_{\mathrm{ty}}(\Gamma) \cup \mathrm{FV}_{\mathrm{ty}}(p)) = \emptyset)\ \frac{\mathcal{E}(\overline{\alpha}), \Gamma \vdash p}{\Gamma \vdash p}\ \text{drop-unused}\mathcal{E}$$

4.2⟨14⟩ The proof of this lemma depends on the theorem (B.2.7)

$$\vdash \mathcal{E}(\alpha)[o/\alpha]$$

that is, $[\dot{=}]_o$ is reflexive, transitive, symmetric, and moreover that $[\dot{=}]_o \mathrel{\dot{=}}_{o \Rightarrow o \Rightarrow o} [\dot{=}]_o$. These are all derivable by unfolding the definitions of $[\dot{=}]$.

4.2⟨15⟩ Then the proof of *drop-unused*$\mathcal{E}$ follows in a similar manner to the proof of *drop-unused*$\mathcal{R}$. We construct the substitution ϕ_o, which substitutes o for each variable in $\overline{\alpha}$. By a similar argument to *drop-unused*$\mathcal{R}$, each of these antecedents can be proven unconditionally, and removed by *ante-subst*.

4.2.4 Symmetry

4.2⟨16⟩ If $\mathcal{E}(\tau)$ then the $[\dot{=}]$ relation is symmetric. This can be show by induction on the type τ (see Subsection B.2.4).

$$\frac{\Gamma \vdash s \mathrel{\dot{=}}_\tau t}{\Gamma, \mathcal{E}(\tau) \vdash t \mathrel{\dot{=}}_\tau s}\ \text{sym}\dot{=}$$

Note that this is stronger than the theorem obtained by translating *sym*, as no $\mathcal{R}$ is needed.

4.2.5 Translation of the Connectives

4.2⟨17⟩ Another intermediate result is that the translation preserves the connectives. This is needed for the rules *select-ax* and *infinity-ax*. These results only hold when the connectives are fully applied. A statement of the results can be found in Figure 4.1, and full proofs are detailed in Subsection B.2.9.

4.2.6 mk-comb$\dot{=}$ and tyapp$\dot{=}$

The rules

$$\frac{\Gamma \vdash s \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} t \qquad \Delta \vdash u \mathrel{\dot{=}}_{\tau_1} v}{\Gamma, \Delta \vdash s\ u \mathrel{\dot{=}}_{\tau_2} t\ v}\ \text{mk-comb}\dot{=} \qquad \frac{\Gamma \vdash s^\bullet \mathrel{\dot{=}}_{\Lambda\alpha.\ \tau} t^\bullet}{\Gamma, \mathcal{E}(\alpha) \vdash s^\bullet\ [{:}\alpha{:}] \mathrel{\dot{=}}_\tau t^\bullet\ [{:}\alpha{:}]}\ \text{tyapp}\dot{=}$$

holds in PHOL. The proof of these rules (B.2.10, B.2.11) follows by unfolding $[\dot{=}]$ on $\tau_1 \Rightarrow \tau_2$ and $\Pi\alpha.\ \tau$.

$$\vdash \top^\bullet =_o \top$$
$$\vdash (p \wedge q)^\bullet =_o p^\bullet \wedge q^\bullet$$
$$\vdash (p \longrightarrow q)^\bullet =_o p^\bullet \longrightarrow q^\bullet$$
$$\vdash (\forall x.\, p)^\bullet =_o (\forall x.\, x \stackrel{\bullet}{=}_\tau x \longrightarrow p^\bullet)$$
$$\vdash (\exists x.\, p)^\bullet =_o (\exists x.\, x \stackrel{\bullet}{=}_\tau x \longrightarrow p^\bullet)$$
$$\vdash (p \vee q)^\bullet =_o p^\bullet \vee q^\bullet$$
$$\vdash \bot^\bullet =_o \bot$$
$$\vdash (\neg p)^\bullet =_o \neg p^\bullet$$
$$\vdash (\forall_{\mathsf{ty}}\alpha.\ p)^\bullet =_o (\forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow p)^\bullet$$

Figure 4.1: Connective Translation Theorems in PHOL

4.2.7 The Substitution Lemma

4.2⟨18⟩ Perhaps the most critical result of the translation is the following lemma about substitution (see also Subsection B.2.12). To state it, we first define the function $\mathfrak{R}$ as

$$\mathfrak{R}(s, \theta_l, \theta_r) \triangleq \{\!\{x[\theta_l] \stackrel{\bullet}{=} x[\theta_r] \mid x \in \mathrm{FV}(s)\}\!\}$$

This function, when given a term s and two substitutions of terms θ_l and θ_r, generates the multi-set of propositions (one for each free variable in s) that assert that for every variable in the term s, the term $x[\theta_l]$ is extensionally partially equivalent to the term $x[\theta_r]$. Note that when these substitutions are empty, this devolves to $\mathcal{R}(s)$.

4.2⟨19⟩ Given this function, the *substitution lemma* is stated as follows

Lemma 4.2.1 (The Substitution Lemma). *If s does not contain $[=]$, except on o and ι, then*

$$\frac{}{\mathcal{E}(s), \mathfrak{R}(s, \theta_l, \theta_r) \vdash s[\theta_l] \stackrel{\bullet}{=}_\tau s[\theta_r]}\ \textit{subst. lemma.}$$

4.2⟨20⟩ What this says is that, if every variable is extensionally partially-equivalent under the substitutions, then any larger term formed from these variables will result in extensionally partially-equivalent terms when the substitutions are applied.

4.2⟨21⟩ Note that the translation $(-)^\bullet$ removes all $[=]$ except those on o and ι, which means that when s is of the form $t^\bullet$, the side condition is satisfied. Further, note that on the empty substitution, $\mathfrak{R}$ devolves to $\mathcal{R}$.

4.2⟨22⟩ The proof proceeds by induction upon the term-structure of s. Firstly, the variable case follows immediately by the assumption of equivalence in $\mathfrak{R}$.

4.2⟨23⟩ In the lambda case, it is easiest to reason backwards from the goal, which breaks into four sub-cases, depending on whether z is in $\mathrm{Vars}(\theta_l)$ or $\mathrm{Vars}(\theta_r)$. If a/z is in θ_l,

we rebind θ_l as $(\theta_l, a/z) \coloneqq \theta_l$. Then $(\lambda z.\ s)[\theta_l, a/z] \equiv (\lambda z.\ s[\theta_l])$ by the definition of substitution; if b/z is in θ_r, then we perform a similar step for θ_r. In all cases, by this rewriting, the goal is then

$$\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r) \vdash (\lambda z.\ s[\theta_l]) \stackrel{\bullet}{=}_{\tau_1 \Rightarrow \tau_2} (\lambda z.\ s[\theta_r])$$

4.2⟨24⟩ By unfolding the definition $[\stackrel{\bullet}{=}]$, and applying $\forall I$ and $\rightarrow I$, we obtain the goal

$$\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r), x \stackrel{\bullet}{=}_{\tau_1} y \vdash (\lambda z.\ s[\theta_l])\ x \stackrel{\bullet}{=}_{\tau_2} (\lambda z.\ s[\theta_r])\ y$$

The key step here is to realise that this is beta-equivalent to

$$\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r), z[\theta_l, x/z] \stackrel{\bullet}{=}_{\tau_1} z[\theta_l, y/z] \vdash s[\theta_l, x/z] \stackrel{\bullet}{=}_{\tau_2} s[\theta_r, y/z]$$

4.2⟨25⟩ Considering the definition of $\mathfrak{R}$

$$\mathfrak{R}(s, (\theta_l, x/z), (\theta_r, y/z)) \equiv z[\theta_l, x/z] \stackrel{\bullet}{=}_{\tau_1} z[\theta_l, y/z]$$

and so we obtain the goal

$$\mathcal{E}(s), \mathfrak{R}(s, (\theta_l, x/z), (\theta_r, y/z)) \vdash s[\theta_l, x/z] \stackrel{\bullet}{=}_{\tau_2} s[\theta_r, y/z]$$

which is provable from the induction hypothesis.

4.2⟨26⟩ The application case is fairly simple. Note that $\mathcal{E}(s\ t)$ is $\mathcal{E}(s) \cup \mathcal{E}(t)$, and $\mathfrak{R}(s\ t, \theta_l, \theta_r)$ is $\mathfrak{R}(s, \theta_l, \theta_r) \cup \mathfrak{R}(t, \theta_l, \theta_r)$. This means the goal is equivalent to

$$\mathcal{E}(s), \mathcal{E}(t), \mathfrak{R}(s, \theta_l, \theta_r), \mathfrak{R}(t, \theta_l, \theta_r) \vdash s[\theta_l]\ t[\theta_l] \stackrel{\bullet}{=}_{\tau} s[\theta_r]\ t[\theta_r]$$

By an application of *mk-comb*$\stackrel{\bullet}{=}$ (4.2.6B.2.10), this reduces to two smaller cases, which are solved by the induction hypothesis.

4.2⟨27⟩ The equality case is tedious, but straightforward. By the precondition, we only need to consider $[=]_o$ and $[=]_\iota$. Abstracting over o and ι with χ, we note that $[=]_\chi$ has no free variables, so the goal to prove is

$$[=]_\chi \stackrel{\bullet}{=}_{o \Rightarrow o \Rightarrow o} [=]_\chi$$

By expanding out $[\stackrel{\bullet}{=}]_{o \Rightarrow o \Rightarrow o}$, this reduces to

$$\forall x_1\ y_1.\ x_1 =_\chi y_1 \longrightarrow (\forall x_2\ y_2.\ x_2 =_\chi y_2 \longrightarrow (x_1 =_\chi x_2) =_o (y_1 =_\chi y_2))$$

which is fairly easy (but tedious) to prove from the introduction rules, *sym*, and *trans*.

4.2⟨28⟩ The $[\stackrel{\bullet}{=}]_\alpha$ case follows almost directly from the definition of $\mathcal{E}(\alpha)$.

4.2⟨29⟩ The type-lambda case can be shown by reasoning backwards from the conclusion. Firstly, note that if α occurs in θ_l or θ_r, we must alpha-rename $\Lambda\alpha.\ s$ to some variable that does not. Having done this, we may interchange substitution and Λ, to obtain the goal

$$\mathcal{E}(\Lambda\alpha.\ s), \Gamma \vdash (\Lambda\alpha.\ s[\theta_l]) \stackrel{\bullet}{=}_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ s[\theta_r]).$$

4.2⟨30⟩ Unfolding the antecedent $[\dot{=}]$ on $\Pi\alpha.$ and applying the introductions to obtain the goal

$$\mathcal{E}(\Lambda\alpha.\ s), \mathcal{E}(\alpha), \Gamma \vdash (\Lambda\alpha.\ s[\theta_l])\ [:\alpha:] \dot{=}_\tau (\Lambda\alpha.\ s[\theta_r])\ [:\alpha:].$$

Nothing that $\mathcal{E}(\Lambda\alpha.\ s), \mathcal{E}(\alpha)$ weakens to $\mathcal{E}(s)$, we can then produce

$$\mathcal{E}(s) \vdash s[\theta_l] \dot{=}_\tau s[\theta_l]$$

which is provable by the induction hypothesis.

4.2⟨31⟩ We prove the type-application case by first noting that $\mathfrak{R}(s\ [:\sigma:], \theta_l, \theta_r)$ is $\mathfrak{R}(s, \theta_l, \theta_r)$ and that $\mathcal{E}(s\ [:\sigma:])$ weakens to $\mathcal{E}(s)$. By these facts, and the definition of substitution, we simplify the goal to

$$\mathcal{E}(s), \mathfrak{R}(s, \theta_l, \theta_r) \vdash s[\theta_l]\ [:\sigma:] \dot{=}_\tau s[\theta_r]\ [:\sigma:]$$

which, after applying *tyinst*$\dot{=}$, follows by induction hypothesis.

4.3 Correctness of Translation

4.3⟨1⟩ As discussed in Section 2.2, the correctness of the translation breaks down into two directions, the validity of the translated theorem, and the non-triviality of the translation.

4.3.1 Translation Validity

4.3⟨2⟩ Recall that to show translation validity, we must show that assuming the source theorem

$$\Gamma \vdash_{\mathrm{PHOL}_{\xi\eta}} s$$

the translated theorem

$$\mathcal{E}(\Gamma^\bullet), \mathcal{E}(s^\bullet), \mathcal{R}(\Gamma^\bullet), \mathcal{R}(s^\bullet), \Gamma^\bullet \vdash_{\mathrm{PHOL}} s^\bullet$$

is provable.

4.3⟨3⟩ The proof of translation validity proceeds by induction over the derivation of the source theorem, and then further by either induction on the type of an equality, or a direct derivation. Full proofs in natural deduction style are given in Subsection C.1.1. This section contains a prose explanation of these derivations.

4.3⟨4⟩ Firstly, recall the rules of $\mathrm{PHOL}_{\xi\eta}$

Structural

$$\frac{}{p \vdash p}\ \text{assm}$$

$$\frac{\Gamma \vdash p}{\Gamma, \Delta \vdash p}\ \text{weaken} \qquad \frac{\Gamma, \Delta, \Delta \vdash p}{\Gamma, \Delta \vdash p}\ \text{contract}$$

Equality

$$\frac{}{\vdash s = s}\ \text{refl} \qquad \frac{\Gamma \vdash s = t \quad \Delta \vdash t = u}{\Gamma, \Delta \vdash s = u}\ \text{trans}$$

$$\frac{\Gamma \vdash s = t \quad \Delta \vdash u = v}{\Gamma, \Delta \vdash s\ u = t\ v}\ \text{mk-comb} \qquad \frac{}{\vdash (\lambda x.\ s)\ t = s[t/x]}\ \text{beta}$$

$$(x \text{ not in } \Gamma)\ \frac{\Gamma \vdash s = t}{\Gamma \vdash (\lambda x.\ s) = (\lambda x.\ t)}\ \text{abs} \qquad (x \text{ not in } t)\ \frac{}{\vdash (\lambda x.\ t\ x) = t}\ \text{eta}$$

$$\frac{\Gamma \vdash s = t}{\Gamma \vdash s\ [{:}\tau{:}] = t\ [{:}\tau{:}]}\ \text{tyapp} \qquad \frac{}{\vdash (\Lambda \alpha.\ s)\ [{:}\sigma{:}] = s[\sigma/\alpha]}\ \text{tybeta}$$

$$(\alpha \text{ not in } \Gamma)\ \frac{\Gamma \vdash s = t}{\Gamma \vdash (\Lambda \alpha.\ s) = (\Lambda \alpha.\ t)}\ \text{tyabs} \qquad (\alpha \text{ not in } f)\ \frac{}{\vdash (\Lambda \alpha.\ f\ [{:}\alpha{:}]) = f}\ \text{tyeta}$$

$$\frac{\Gamma \vdash s =_o t \quad \Delta \vdash s}{\Gamma, \Delta \vdash t}\ \text{eq-mp} \qquad \frac{\Gamma, q \vdash p \quad \Delta, p \vdash q}{\Gamma, \Delta \vdash p =_o q}\ \text{deduct-antisym}$$

Axioms

$$(x \text{ not in } p)\ \frac{}{\vdash p\ s \longrightarrow p\ (\varepsilon x.\ p\ x)}\ \text{select-ax} \qquad \frac{}{\vdash \exists (f : \iota \Rightarrow \iota).\ \text{inj}\ f \wedge \neg\, \text{onto}\ f}\ \text{infinity-ax}$$

4.3.2 The Structural Rules

4.3⟨5⟩ The structural rules of *assm*, *weaken*, and *contract* are very simple direct proofs from the same rules in PHOL.

4.3.3 refl

4.3⟨6⟩ The admissibility of the translated version of *refl*

$$\frac{}{\mathcal{E}(\tau), \mathcal{E}(s^\bullet), \mathcal{R}(s^\bullet) \vdash s^\bullet \doteq_\tau s^\bullet}\ \text{refl}\bullet$$

follows from the substitution lemma Subsection 4.2.7 (B.2.12), as on an empty substitution, the antecedents become $\mathcal{E}(\tau), \mathcal{E}(s^\bullet), \mathcal{R}(s^\bullet)$.

4.3.4 trans

4.3⟨7⟩ The translated rule for *trans* is an extremely fiddly proof. To begin, we prove a generalised form of the translated rule: *trans*$\dot{=}$

$$\frac{\mathcal{E}(\tau), \mathcal{E}(\Gamma, s, t), \mathbf{U}\mathcal{R}_t, \Gamma \vdash s \mathrel{\dot{=}}_\tau t \qquad \mathcal{E}(\tau), \mathcal{E}(\Delta, s, t), \mathbf{U}\mathcal{R}_t, \Delta \vdash t \mathrel{\dot{=}}_\tau u}{\mathcal{E}(\tau), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash s \mathrel{\dot{=}}_\tau u} \text{ trans}\dot{=}$$

where $\mathbf{U}\mathcal{R}_{t\bullet}$ is defined as

$$\mathbf{U}\mathcal{R}_{t\bullet} \triangleq \mathcal{R}(\mathrm{FV}(t^\bullet) - \mathrm{FV}(\Gamma, \Delta, s^\bullet, t^\bullet)).$$

The proof proceeds by induction on τ.

4.3⟨8⟩ On the basic types o and ι, the proof is by unfolding the definition of $\dot{=}$, applying the *trans* of PHOL, then removing the unnecessary antecedents using *drop-unused*$\mathcal{R}$ and then *drop-unused*$\mathcal{E}$.

4.3⟨9⟩ The hardest case is the proof for function types $\tau_1 \Rightarrow \tau_2$. Reasoning from the hypotheses, by unfolding the definition of $[\dot{=}]$ and eliminating the connectives, we can derive

$$\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, s), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Gamma, x \mathrel{\dot{=}}_{\tau_1} y \vdash s\ x \mathrel{\dot{=}}_{\tau_2} t\ y$$

and

$$\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Delta, u), \Delta, y \mathrel{\dot{=}}_{\tau_1} y \vdash t\ y \mathrel{\dot{=}}_{\tau_2} u\ y$$

4.3⟨10⟩ Applying the induction hypothesis (and contracting) produces

$$\mathcal{E}(\tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta, x \mathrel{\dot{=}}_{\tau_1} y, y \mathrel{\dot{=}}_{\tau_1} y \vdash s\ x \mathrel{\dot{=}}_{\tau_2} u\ y$$

which is nearly in the correct form to produce $s \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} u$; however, there is a lingering $y \mathrel{\dot{=}}_{\tau_1} y$ antecedent which must be removed.

4.3⟨11⟩ To remove this antecedent, we use *ante-subst* with the following lemma.

$$\mathcal{E}(\tau_1), x \mathrel{\dot{=}}_{\tau_1} y \vdash y \mathrel{\dot{=}}_{\tau_1} y.$$

This lemma can be derived by using the induction hypothesis on $y \mathrel{\dot{=}}_{\tau_1} x$ and $x \mathrel{\dot{=}}_{\tau_1} y$, which are derivable by contraction and *sym*$\dot{=}$ (4.2.4). This step is what necessitates the generalised rule.

4.3⟨12⟩ After removing this antecedent, we now have the theorem

$$\mathcal{E}(\tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta, x \mathrel{\dot{=}}_{\tau_1} y \vdash s\ x \mathrel{\dot{=}}_{\tau_2} u\ y$$

from which it is simple to produce the required conclusion by the application of the introduction rules and the definition of $[\dot{=}]$.

4.3⟨13⟩ For the polymorphic function types, reasoning for the assumptions, by unfolding $[\dot{=}]$ and eliminating the connectives, we obtain the theorems

$$\mathcal{E}(t\ [:\alpha:]), \mathbf{U}\mathcal{R}_{t\ [:\alpha:]}, \mathcal{E}(\tau), \mathcal{E}(\Gamma, s\ [:\alpha:]), \Gamma, \mathcal{E}(\alpha) \vdash s\ [:\alpha:] \dot{=}_\tau t\ [:\alpha:]$$

and

$$\mathcal{E}(t\ [:\alpha:]), \mathbf{U}\mathcal{R}_{t\ [:\alpha:]}, \mathcal{E}(\tau), \mathcal{E}(\Delta, u\ [:\alpha:]), \Delta, \mathcal{E}(\alpha) \vdash t\ [:\alpha:] \dot{=}_\tau u\ [:\alpha:].$$

Applying the inductive hypothesis, we obtain

$$\mathcal{E}(t\ [:\alpha:]), \mathbf{U}\mathcal{R}_{t\ [:\alpha:]}, \mathcal{E}(\tau), \mathcal{E}(\Gamma, s\ [:\alpha:]), \mathcal{E}(\Delta, u\ [:\alpha:]), \Delta, \Gamma, \mathcal{E}(\alpha) \vdash s\ [:\alpha:] \dot{=}_\tau u\ [:\alpha:]$$

which, by connective introductions and the definition of $\dot{=}$ produces the required conclusion.

4.3⟨14⟩ For the type-variable case, reasoning forward we use the hypotheses

$$\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\alpha), \mathcal{E}(\Gamma, s), \Gamma \vdash s \dot{=}_\alpha t$$

and

$$\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\alpha), \mathcal{E}(\Delta, u), \Delta \vdash t \dot{=}_\alpha u$$

to discharge the assumptions of

$$\mathcal{E}(\alpha) \vdash \forall x\ y\ z.\, x \dot{=}_\alpha y \longrightarrow y \dot{=}_\alpha z \longrightarrow x \dot{=}_\alpha z$$

.

4.3⟨15⟩ Cleaning up the antecedents with *contract*, this results in the theorem

$$\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\alpha), \mathcal{E}(\Gamma, \Delta, u), \mathcal{E}(\Gamma, \Delta, s), \Gamma, \Delta \vdash s \dot{=}_\alpha u.$$

Using *drop-unused*$\mathcal{R}$, we can reduce $\mathbf{U}\mathcal{R}_t$ to $\mathcal{E}(\mathbf{U}\mathcal{R}_t)$. This is absorbed by $\mathcal{E}(t)$, which can be reduced to $\mathbf{U}\mathcal{E}_t$ by *contract*, which may be removed by *drop-unused*$\mathcal{E}$. This results in the conclusion.

4.3.5 mk-comb

4.3⟨16⟩ The translation of *mk-comb* is a direct proof from the translated terms. The translated rule is

$$\frac{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \dot{=}_{\tau_1 \Rightarrow \tau_2} t^\bullet \qquad \mathcal{E}(\tau_1), \mathcal{E}(\Delta^\bullet, u^\bullet, v^\bullet), \mathcal{R}(\Delta^\bullet, u^\bullet, v^\bullet), \Delta^\bullet, \vdash u^\bullet \dot{=}_{\tau_1} v^\bullet}{\mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, (s\ u)^\bullet, (t\ v)^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, (s\ u)^\bullet, (t\ v)^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash (s\ u)^\bullet \dot{=}_{\tau_2} (t\ v)^\bullet}$$

4.3⟨17⟩ The proof (C.1.7) proceeds by using the lemma *mk-comb*$\dot{=}$ to obtain the theorem

$$\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, s^\bullet, u^\bullet, t^\bullet, v^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, s^\bullet, u^\bullet, t^\bullet, v^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash s^\bullet\ u^\bullet \dot{=}_{\tau_2} t^\bullet\ v^\bullet.$$

The only significant difference between this theorem and the desired conclusion is the presence of $\mathcal{E}(\tau_1 \Rightarrow \tau_2)$. This assumption may contain more antecedents than are present in the conclusion. These are removed using ***drop-unused$\mathcal{E}$*** (B.2.8), which instantiates any unused type-variable with o, for which $\mathcal{E}$ holds tautologically.

4.3.6 beta

4.3⟨18⟩ The translated rule for ***beta*** is

$$\frac{}{\mathcal{E}(\tau), \mathcal{E}(((\lambda x.\ s)\ t)^\bullet, (s[t/x])^\bullet), \mathcal{R}(((\lambda x.\ s)\ t)^\bullet, (s[t/x])^\bullet), \vdash ((\lambda x.\ s)\ t)^\bullet \doteq_\tau (s[t/x])^\bullet}$$

4.3⟨19⟩ The proof of correctness of translation (see also Subsection C.1.8) proceeds as follows. Reasoning in reverse, by the definition of $\mathcal{R}$ and $\mathcal{E}$, $((\lambda x.\ s)\ t)^\bullet$ is equivalent to $(\lambda x.\ s^\bullet)\ t^\bullet$ by definition, and $(s[t/x])^\bullet$ is equivalent to $s^\bullet[t^\bullet/x]$ by ***subst$\bullet$***. Then $(s[t/x])^\bullet$ can be expanded by ***rewrite*** and ***beta*** to $(\lambda x.\ s^\bullet)\ t^\bullet$ which produces (with weakening)

$$\mathcal{E}(\tau), \mathcal{E}((\lambda x.\ s^\bullet)\ t^\bullet), \mathcal{R}((\lambda x.\ s^\bullet)\ t^\bullet) \vdash (\lambda x.\ s^\bullet)\ t^\bullet \doteq_\tau (\lambda x.\ s^\bullet)\ t^\bullet$$

which is provable by ***refl$\bullet$***

4.3.7 abs

The translated rule is

$$(z \text{ not in } \Gamma^\bullet)\ \frac{\mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_{\tau_2} t^\bullet}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma^\bullet, (\lambda z.\ s)^\bullet, (\lambda z.\ t)^\bullet), \mathcal{R}(\Gamma^\bullet, (\lambda z.\ s)^\bullet, (\lambda z.\ t)^\bullet), \Gamma^\bullet \vdash (\lambda z.\ s)^\bullet \doteq_{\tau_1 \Rightarrow \tau_2} (\lambda z.\ t)^\bullet}\ \text{abs}\bullet$$

4.3⟨20⟩ Reasoning in a forward direction, we apply ***inst*** to the assumption to obtain

$$\mathcal{E}(\tau_2)[x/z], \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet)[x/z], \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet)[x/z], \Gamma^\bullet[x/z] \vdash s^\bullet[x/z] \doteq_{\tau_2} t^\bullet[x/z]$$

and then use the definition of $\mathcal{E}$ and $\mathcal{R}$ to simplify this to

$$\mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet[x/z], t^\bullet[x/z]), \mathcal{R}(\Gamma^\bullet[x/z], s^\bullet[x/z], t^\bullet[x/z]), \Gamma^\bullet[x/z] \vdash s^\bullet[x/z] \doteq_{\tau_2} t^\bullet[x/z].$$

Noting that z is not in $\Gamma^\bullet$ (as z is not in Γ), this further simplifies to

$$\mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet[x/z], t^\bullet[x/z]), \mathcal{R}(\Gamma^\bullet, s^\bullet[x/z], t^\bullet[x/z]), \Gamma^\bullet \vdash s^\bullet[x/z] \doteq_{\tau_2} t^\bullet[x/z]$$

which is, by ***beta*** and ***contract*** is

$$\mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, (\lambda z.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\lambda z.\ t^\bullet)), \Gamma^\bullet, x \doteq x \vdash (\lambda z.\ s^\bullet)\ x \doteq_{\tau_2} (\lambda z.\ t^\bullet)\ x.$$

4.3⟨21⟩ The next step is to derive the theorem

$$\mathcal{E}(\tau_1), \mathcal{E}(\tau_2), \mathcal{E}(\lambda x.\ t^\bullet), \mathcal{R}(\lambda x.\ t^\bullet), x \doteq_{\tau_1} y \vdash (\lambda z.\ t^\bullet)\ x \doteq_{\tau_2} (\lambda z.\ t^\bullet)\ y.$$

This follows from ***mk-comb***$\dot{=}$ and ***refl***$\bullet$ on $(\lambda z.\ t^\bullet)$.

4.3⟨22⟩ Then by ***trans***$\dot{=}$ we can obtain the theorem

$$\mathcal{E}(\tau_1), \mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, (\lambda z.\ s^\bullet), (\lambda z.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\lambda z.\ s^\bullet), (\lambda z.\ t^\bullet)), \Gamma^\bullet, x \mathrel{\dot{=}}_{\tau_1} y, x \mathrel{\dot{=}}_{\tau_1} x \vdash (\lambda z.\ s^\bullet)\ x \mathrel{\dot{=}}_{\tau_2} (\lambda z.\ t^\bullet)\ y.$$

Before continuing we must eliminate $x \mathrel{\dot{=}}_{\tau_1} x$, which can be achieved—in a similar manner to the proof of trans—by proving the theorem

$$\mathcal{E}(\tau_1), x \mathrel{\dot{=}}_{\tau_1} y \vdash x \mathrel{\dot{=}}_{\tau_1} x$$

using ***trans***$\dot{=}$, ***sym***$\dot{=}$, and then applying ***ante-subst.***

4.3⟨23⟩ Finally, by appeal to the connective introduction rules and the definitions of $\mathcal{E}$ and $\dot{=}$.

$$\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma^\bullet, (\lambda z.\ s^\bullet), (\lambda z.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\lambda z.\ s^\bullet), (\lambda z.\ t^\bullet)), \Gamma^\bullet \vdash (\lambda z.\ s^\bullet) \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} (\lambda z.\ t^\bullet)$$

which is the desired conclusion.

4.3.8 eta

4.3⟨24⟩ The translated rule for ***eta*** is

$$(x \text{ not in } t^\bullet)\ \frac{}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}((\lambda x.\ t^\bullet\ x), t^\bullet), \mathcal{R}((\lambda x.\ t^\bullet\ x), t^\bullet) \vdash (\lambda x.\ t^\bullet\ x) \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} t^\bullet}\ \text{eta}\bullet$$

The proof of the translated rule for eta proceeds by direct proof.

4.3⟨25⟩ Reasoning from the goal, we unfold the definition of $[\dot{=}]$, apply the connective introduction rules, and weaken to obtain

$$\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(t^\bullet), \mathcal{R}(t^\bullet), y_1 \mathrel{\dot{=}}_{\tau_1} y_2 \vdash (\lambda x.\ t^\bullet\ x)\ y_1 \mathrel{\dot{=}}_{\tau_2} t^\bullet\ y_2$$

By ***beta***, $(\lambda x.\ t^\bullet\ x)\ y_1$ is equivalent to $t^\bullet\ y_1$, which can be broken up by ***mk-comb***$\dot{=}$ (4.2.6, B.2.10) into the goals

$$y_1 \mathrel{\dot{=}}_{\tau_1} y_2 \vdash y_1 \mathrel{\dot{=}}_{\tau_1} y_2$$

which follows by assumption, and

$$\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(t^\bullet), \mathcal{R}(t^\bullet), y_1 \mathrel{\dot{=}}_{\tau_1} y_2 \vdash t^\bullet \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} t^\bullet$$

which follows by ***refl***$\bullet$

4.3.9 tyapp

4.3⟨26⟩ The translated rule for ***tyapp*** is

$$\frac{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet, \vdash s^\bullet \doteq_{\Pi\alpha.\ \tau} t^\bullet}{\mathcal{E}(\tau), \mathcal{E}(\Gamma^\bullet, (s\ [:\alpha:])^\bullet, (t\ [:\alpha:])^\bullet), \mathcal{R}(\Gamma^\bullet, (s\ [:\alpha:])^\bullet, (t\ [:\alpha:])^\bullet), \Gamma^\bullet \vdash (s\ [:\alpha:])^\bullet \doteq_\tau (t\ [:\alpha:])^\bullet}$$

4.3⟨27⟩ To prove this, we use the generalised ***tyapp***$\doteq$ (B.2.11)

$$\frac{\Gamma \vdash s^\bullet \doteq_{\Lambda\alpha.\ \tau} t^\bullet}{\mathcal{E}(\alpha), \Gamma \vdash s^\bullet\ [:\alpha:] \doteq_\tau t^\bullet\ [:\alpha:]}\ \text{tyapp}\doteq$$

which follows by unfolding the definition of $\doteq$ on polymorphic functions, and performing routine elimination steps on the connectives.

4.3⟨28⟩ The proof of the translated rule (C.1.11), is fairly simple. Firstly, application of ***tyapp***$\doteq$ to the hypothesis produces the theorem

$$\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\alpha), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet\ [:\alpha:] \doteq_\tau t^\bullet\ [:\alpha:].$$

The next, slightly subtle, step is realising that $\mathcal{E}(\Pi\alpha.\ \tau)$ implies $\mathcal{E}(\tau)$. By cases: if $\mathrm{FV}_{\mathrm{ty}}(\tau)$ contains α, then we may use ***weaken*** to generate these antecedents; if it does not, then $\mathcal{E}(\Pi\alpha.\ \tau)$ is equal to $\mathcal{E}(\tau)$.

4.3⟨29⟩ From this, the final steps involve manipulation of the $\mathcal{E}$ and $\mathcal{R}$ antecedents and rewriting by the definition of $\bullet$.

4.3.10 tybeta

4.3⟨30⟩ The translated rule for ***beta*** is

$$\frac{}{\mathcal{E}(\tau), \mathcal{E}(((\lambda x.\ s)\ t), s[t/x]), \mathcal{R}(((\lambda x.\ s)\ t), s[t/x]), \vdash ((\lambda x.\ s)\ t)^\bullet \doteq_\tau (s[t/x])^\bullet}$$

4.3⟨31⟩ The proof of the admissibility of this rule is as follows (see also Subsection C.1.12). Reasoning in reverse, by the definition of $\mathcal{R}$ and $\mathcal{E}$, $((\Lambda\alpha.\ s^\bullet)\ [:\sigma:])^\bullet$ is equivalent to $(\Lambda\alpha.\ s^\bullet)\ [:\sigma:]$ by definition, and $(s[\sigma/\alpha])^\bullet$ is equivalent to $s^\bullet[\sigma/\alpha]$ by ***tysubst***$\bullet$. Then $(s[\sigma/\alpha])^\bullet$ can be expanded by ***rewrite*** with ***tybeta*** to $(\Lambda\alpha.\ s^\bullet)\ [:\sigma:]$ which produces (with weakening)

$$\mathcal{E}(\tau), \mathcal{E}((\Lambda\alpha.\ s)\ [:\sigma:]), \mathcal{R}((\Lambda\alpha.\ s)\ [:\sigma:]) \vdash (\Lambda\alpha.\ s^\bullet)\ [:\sigma:] \doteq_\tau (\Lambda\alpha.\ s^\bullet)\ [:\sigma:]$$

which is provable by ***refl***$\bullet$.

4.3.11 tyabs

4.3⟨32⟩ The translated rule for *tyabs* is

$$(\alpha \text{ not in } \Gamma^\bullet)\ \frac{\mathcal{E}(\tau), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_\tau t^\bullet}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s)^\bullet, (\Lambda\alpha.\ t)^\bullet), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s)^\bullet, (\Lambda\alpha.\ t)^\bullet), \Gamma^\bullet \vdash (\Lambda\alpha.\ s)^\bullet \doteq_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)^\bullet}\ \text{tyabs}\bullet$$

4.3⟨33⟩ Starting from the hypothesis, first note the following argument: the antecedents $\mathcal{E}(s^\bullet)$ implies the antecedents $\mathcal{E}(\Lambda\alpha.\ s^\bullet), \mathcal{E}(\alpha)$ (and similarly for t). The proof for this is as follows: If s contains the free variable α, then $\mathcal{E}(s^\bullet)$ implies $\mathcal{E}((\Lambda\alpha.\ s)^\bullet), \mathcal{E}(\alpha)$ by contraction. If s does not contain the free variable α, then $\mathcal{E}((\Lambda\alpha.\ s)^\bullet), \mathcal{E}(\alpha)$ follows by weakening.

4.3⟨34⟩ By this argument, and by further noting $\mathcal{R}(s^\bullet)$ is the same as $(\Lambda\alpha.\ s^\bullet)$, we may derive

$$\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\alpha), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \Gamma^\bullet \vdash s^\bullet \doteq_\tau t^\bullet$$

4.3⟨35⟩ From this, by $\to I$ and $\forall_{\mathsf{ty}} I$, we can derive

$$\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \Gamma^\bullet \vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ s^\bullet)\ [:\alpha:] \doteq_\tau (\Lambda\alpha.\ t^\bullet)\ [:\alpha:]$$

There is one subtlety to this, which is that $\forall_{\mathsf{ty}} I$ requires α free in all the antecedents; this reduces to the fact that 'α is not it Γ' from the definitions of $\mathcal{E}$ and $\mathcal{R}$.

4.3⟨36⟩ Having done this, by the definition of $[\doteq]$ on polymorphic types and the definition of $\bullet$, we have produced the required conclusion. A full proof can be found in Subsection C.1.13.

4.3.12 tyeta

4.3⟨37⟩ The translated rule for *tyeta* is

$$(\alpha \text{ not in } f^\bullet)\ \frac{}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]), f^\bullet), \mathcal{R}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]), f^\bullet) \vdash (\Lambda\alpha.\ f^\bullet\ [:\alpha:]) \doteq_{\Pi\alpha.\ \tau} f^\bullet}\ \text{tyeta}$$

4.3⟨38⟩ Firstly, note that, because α does not occur in f, it also does not occur in $f^\bullet$, because $(-)^\bullet$ preserves type variables.

4.3⟨39⟩ Working backwards from the conclusion, $\mathcal{R}(\Lambda\alpha.\ f^\bullet\ [:\alpha:])$ can be simplified to $\mathcal{R}(f^\bullet)$, and $\mathcal{E}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]))$ can be simplified to $\mathcal{E}(f^\bullet)$, because α does not occur in $f^\bullet$. Simplifying the antecedents with weakening, we obtain the goal

$$\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(f^\bullet), \mathcal{R}(f^\bullet) \vdash (\Lambda\alpha.\ f^\bullet\ [:\alpha:]) \doteq_{\Pi\alpha.\ \tau} f^\bullet$$

4.3⟨40⟩ Unfolding the definition of $[\dot{=}]$ on polymorphic functions, and reducing the connectives with $\to$I and $\forall_{\text{ty}}$I, we further reduce the goal to

$$\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(f^\bullet), \mathcal{R}(f^\bullet), \mathcal{E}(\alpha) \vdash (\Lambda\alpha.\ f^\bullet\ [{:}\alpha{:}])\ [{:}\alpha{:}] \dot{=}_\tau f^\bullet\ [{:}\alpha{:}]$$

The $\forall_{\text{ty}}I$ is safe, as α does not occur in $\mathcal{E}(\Pi\alpha.\ \tau)$, and not in $f^\bullet$. By *tybeta* and the fact that α does not occur in $f^\bullet$, this is equivalent to

$$\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(f^\bullet), \mathcal{R}(f^\bullet), \mathcal{E}(\alpha) \vdash f^\bullet\ [{:}\alpha{:}] \dot{=}_\tau f^\bullet\ [{:}\alpha{:}]$$

4.3⟨41⟩ Finally, using the facts that $\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\alpha)$ produces $\mathcal{E}(\tau), \mathcal{E}(\alpha)$ by contraction, $\mathcal{E}(f), \mathcal{E}(\alpha)$ is equivalent to $\mathcal{E}(f\ [{:}\alpha{:}])$, and $\mathcal{R}(f)$ is the same as $\mathcal{R}(f\ [{:}\alpha{:}])$, we may rewrite the antecedents into the correct form to apply ***refl•***, which completes the proof.

4.3.13 eq-mp

The translated rule for *eq-mp* is

$$\frac{\mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \dot{=}_o t^\bullet \qquad \mathcal{E}(\Delta^\bullet, s^\bullet), \mathcal{R}(\Delta^\bullet, s^\bullet), \Delta^\bullet \vdash s^\bullet}{\mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{eq-mp}\bullet$$

4.3⟨42⟩ Reasoning from the hypothesis, as $[\dot{=}]_o$ is just $[=]_o$ it is quite simple to obtain—by application of *eq-mp* and *contract*—the theorem

$$\mathcal{E}(s^\bullet), \mathcal{R}(s^\bullet), \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet$$

However, this has the antecedents $\mathcal{E}(s^\bullet)$ and $\mathcal{R}(s^\bullet)$, which must be removed to show the conclusion of the rule.

4.3⟨43⟩ Recalling the *drop-unused*$\mathcal{R}$ lemma (9, B.2.6) we divide $\mathcal{R}(s^\bullet)$ into the propositions also found in $\mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet)$, and those which aren't. The former can be contracted away, leaving only the $\mathcal{R}$-antecedents of variable which occur only in $s^\bullet$; call these $\mathbf{U}\mathcal{R}_s\bullet$.

4.3⟨44⟩ Applying *drop-unused*$\mathcal{R}$, we remove these antecedents, but are left with $\mathcal{E}(\mathbf{U}\mathcal{R}_s\bullet)$ (by the rule) and $\mathcal{E}(s^\bullet)$ (from earlier). Doing a similar move to before, we can split these $\mathcal{E}$ antecedents into those shared with other antecendents, which we *contract* away, and those which aren't.

4.3⟨45⟩ Finally, we remove these antecedents using *drop-unused*$\mathcal{E}$ (12, B.2.8), which produces the desired conclusion.

4.3.14 deduct-antisym

4.3⟨46⟩ The translated rule for ***deduct-antisym*** is

$$\frac{\mathcal{E}(\Gamma^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Gamma^\bullet, p^\bullet, q^\bullet), \Gamma^\bullet, q^\bullet \vdash p^\bullet \qquad \mathcal{E}(\Delta^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Delta^\bullet, p^\bullet, q^\bullet), \Delta^\bullet, p^\bullet \vdash q^\bullet}{\mathcal{E}(\Gamma^\bullet, \Delta^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, p^\bullet, q^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash p^\bullet =_o q^\bullet}\ \text{deduct-antisym}\bullet$$

The proof of this is quite simple, and follows directly from ***deduct-antisym*** in PHOL, ***contract***, and the definition of $[\dot{=}]_o$.

The Axioms 4.3⟨47⟩ The axioms are quite simple. The translation of ***select-ax*** is proven simply by applying the connective translation lemmas (4.2.5, B.2.9). The translation of ***infinity-ax*** is only slightly more complex; the proof requires realising that $\vdash (f \dot{=}_{\iota \Rightarrow \iota} f)$ and $\top \longrightarrow p = p$, which simplifies the connective translation lemmas for $[\forall]$ and $[\exists]$.

4.3.15 Non Triviality

4.3⟨48⟩ To show that the translation is non-trivial, we need to show that if

$$\mathcal{E}(\Gamma^\bullet), \mathcal{E}(s^\bullet), \mathcal{R}(\Gamma^\bullet), \mathcal{R}(s^\bullet), \Gamma^\bullet \vdash_{\text{PHOL}} s^\bullet$$

holds in PHOL, then *any* theorem it could have been translated from

$$\Gamma \vdash_{\text{PHOL}_{\xi\eta}} s$$

holds in $\text{PHOL}_{\xi\eta}$.

4.3⟨49⟩ The proof of this is conceptually quite simple. Firstly, we define a 'reverse translation' $(-)^\circ$

$$\begin{array}{lcl}
x^\circ & \triangleq & x \\
(\lambda x.\ s)^\circ & \triangleq & \lambda x.\ s^\circ \\
(s\ t)^\circ & \triangleq & s^\circ\ t^\circ \\
(\Lambda\alpha.\ s)^\circ & \triangleq & \Lambda\alpha.\ s^\circ \\
(s\ [:\tau:])^\circ & \triangleq & s^\circ\ [:\tau:] \\
([=]_\tau)^\circ & \triangleq & [=]_\tau \\
([\dot{=}]_\alpha)^\circ & \triangleq & [=]_\alpha \\
([\forall]_\tau)^\circ & \triangleq & [\forall]_\tau \\
([\forall_{\mathsf{ty}}])^\circ & \triangleq & [\forall_{\mathsf{ty}}]
\end{array}$$

This function maps PHOL terms back into $\text{PHOL}_{\xi\eta}$, turning equivalent terms to equivalent terms. On the new symbols $[\forall]_\tau$ and $[\forall_{\mathsf{ty}}]$, it maps them to their definitions in $\text{PHOL}_{\xi\eta}$, and on $[\dot{=}]_\alpha$—the only $[\dot{=}]$ which remains after translation—it maps it back to the extensional equality $[=]_\alpha$.

4.3⟨50⟩ The following theorems hold (proofs are in Section C.2)

- In $\mathrm{PHOL}_{\xi\eta}$, $s^{\bullet\circ} = s$ (this is proven by induction on s);
- If $\Gamma \vdash_{\mathrm{PHOL}} p$ is a theorem of PHOL, then $\Gamma^{\circ} \vdash_{\mathrm{PHOL}_{\xi\eta}} p^{\circ}$ is a theorem of $\mathrm{PHOL}_{\xi\eta}$.
- In $\mathrm{PHOL}_{\xi\eta}$, $\mathcal{R}(\overline{x})^{\circ}$ and $\mathcal{E}(\overline{x})^{\circ}$ are tautologies; and

From these three theorems, non-triviality of the translation can be derived as follows

$$\dfrac{\dfrac{\dfrac{}{\vdash^{*} \mathcal{E}(\Gamma^{\bullet})^{\circ}, \mathcal{E}(s^{\bullet})^{\circ}, \mathcal{R}(\Gamma^{\bullet})^{\circ}, \mathcal{R}(s^{\bullet})^{\circ}}\ \mathcal{R}\circ + \mathcal{E}\circ \qquad \dfrac{\mathcal{E}(\Gamma^{\bullet}), \mathcal{E}(s^{\bullet}), \mathcal{R}(\Gamma^{\bullet}), \mathcal{R}(s^{\bullet}), \Gamma^{\bullet} \vdash_{\mathrm{PHOL}} s^{\bullet}}{\mathcal{E}(\Gamma^{\bullet})^{\circ}, \mathcal{E}(s^{\bullet})^{\circ}, \mathcal{R}(\Gamma^{\bullet})^{\circ}, \mathcal{R}(s^{\bullet})^{\circ}, \Gamma^{\bullet\circ} \vdash_{\mathrm{PHOL}_{\xi\eta}} s^{\bullet\circ}}\ \text{subsystem}}{\Gamma^{\bullet\circ} \vdash_{\mathrm{PHOL}_{\xi\eta}} s^{\bullet\circ}}\ \text{ante-subst}^{*}}{\Gamma \vdash_{\mathrm{PHOL}_{\xi\eta}} s}\ \text{rewrite + translate-reverse-eq-id}$$

4.3.16 Conclusion

4.3⟨51⟩ With these two proofs, the correctness and non-triviality of the translation has been shows. This means that proofs in $\mathrm{PHOL}_{\xi\eta}$, including those that use extensional rules such as *eta* and *abs*, can be translated into equivalent theorems in PHOL with the $[\overset{\bullet}{=}]_{\alpha}$ symbol—an intensional logic.

Chapter 5

Conclusion and Future Work

5.0⟨1⟩ This report has presented a translation from extensional polymorphic HOL to intensional polymorphic HOL, and proven its correctness and non-triviality. This extends the state of the art in extensionality translations for higher-order logic [Gandy, 1956; Rizkallah, 2009; Brown and Rizkallah, 2013], which have only handled higher-order logics without polymorphism or type-variables.

5.0⟨2⟩ Proofs of the consistency and completeness of the logics used in this report are future work. I expect them to be consistent and complete, as they are minor deviations from HOL2P—the removal of some types, the addition of the structural rules, *tyeta*, and the connectives $[\forall]$, and $[\forall_{\mathsf{ty}}]$, and a slight change to the rule *deduct-antisym*; however, confidence is not certainty. These proofs would most likely require the use of Heinkin semantics [Henkin, 1950], similar to the consistency proofs of HOL2P [Völker, 2007].

5.0⟨3⟩ Despite proving that the translation is correct—that is, the translated proof rules that define the logic are admissible—and non-trivial—that is, provable translated theorems must have come from provable theorems from the source logic—the translation is not a true embedding into the logic PHOL because the translated terms include the additional symbol $[\dot{=}]_\alpha$. This symbol is needed to define the extensional partial-equivalence by induction on types.

5.0⟨4⟩ If the logic was extended with a function with type-dependent evaluation (that is, reduced to different values depending on the type it was given), the symbol $[\dot{=}]_\alpha$ could instead be defined in the logic. (Whether such a logic could even be consistent is another question that would need to be answered.) Further in this research direction, if a function which allowed term-dependent evaluation was added to the object-logic (for an example of such a term, see the intensional lambda calculus of Jay [2019]), it is possible the entire translation could be embedded into the logic.

Extension to Other Logics

5.0⟨5⟩ This work has extended the extensionality translation to a polymorphic HOL based on HOL2P [Völker, 2007], which adds consistent type-polymorphism to higher-order logic by using ranks. HOL-Omega [Harrison, 2009] is an extension of HOL2P which uses natural-number ranks, which increases the expressivity of the logic. I conjecture that the structure of the translation presented in this work should be extensible to HOL-Omega, but this would need to be proven.

5.0⟨6⟩ Extending the translation to type-constructors or type-operators, which are present in HOL2P—but were not included in the polymorphic HOL of this report for simplicity of the logic—is another possible extension.

5.0⟨7⟩ A critical assumption in this translation—and indeed in the translations of Gandy [1956] and Rizkallah [2009]—is the non-emptiness of types. This allows the definition of the reflexive term Z_τ which is critical to the *drop-unused*$\mathcal{R}$ lemma. An interesting question is if there is a way to extend the translation to systems with empty types, or if such a translation is impossible.

Prove the Translation Correct in a Theorem Prover

5.0⟨8⟩ Another line of further research is to prove this or other translation results in a theorem prover. This would require an embedding of the two polymorphic higher-order logics in a theorem prover, which would be a significant undertaking. Similar work in embedding HOL Light into itself has been performed by Kumar et al. [2016].

Bibliography

T. Altenkirch. 1999. Extensional Equality in Intensional Type Theory. In *Symposium on Logic in Computer Science*. IEEE Computer Society, Los Alamitos, CA, USA, 412. https://doi.org/10.1109/LICS.1999.782636

Hendrik Pieter Barendregt. 1981. *The lambda calculus: its syntax and semantics*. North-Holland Pub. Co., Amsterdam ; New York : New York.

Gilles Barthe, Venanzio Capretta, and Olivier Pons. 2003. Setoids in type theory. *Journal of Functional and Logic Programming* 13 (2003), 261–293. https://hal.archives-ouvertes.fr/hal-01124972 Special Issue on Logical Frameworks and Metalanguages.

Chad E. Brown and Christine Rizkallah. 2013. From Classical Extensional Higher-Order Tableau to Intuitionistic Intentional Natural Deduction. In *Third International Workshop on Proof Exchange for Theorem Proving, PxTP 2013, Lake Placid, NY, USA, June 9-10, 2013*. 27–42. http://www.easychair.org/publications/paper/141242

Alonzo Church. 1940. A Formulation of the Simple Theory of Types. *The Journal of Symbolic Logic* 5, 2 (1940), 56–68. http://www.jstor.org/stable/2266170

Thierry Coquand. 1995. A new paradox in type theory. In *Logic, Methodology and Philosophy of Science IX*, Dag Prawitz, Brian Skyrms, and Dag Westerståhl (Eds.). Studies in Logic and the Foundations of Mathematics, Vol. 134. Elsevier, 555 – 570. https://doi.org/10.1016/S0049-237X(06)80062-5

R. O. Gandy. 1956. On the Axiom of Extensionality–Part I. *J. Symbolic Logic* 21, 1 (03 1956), 36–48. https://projecteuclid.org:443/euclid.jsl/1183732302

Herman Geuvers. 2007. (In)consistency of Extensions of Higher Order Logic and Type Theory. In *Types for Proofs and Programs*, Thorsten Altenkirch and Conor McBride (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 140–159. https://doi.org/10.1007/978-3-540-74464-1_10

Valery Glivenko. 1929. Sur quelques points de la logique de M. Brouwer. (1929).

Kurt Gödel. 1933. On intuitionistic arithmetic and number theory. In *Kurt Gödel: Collected Works*, S. Feferman, John W. Dawson, Stephen C. Kleene Jr., G. Moore, R. Solovay, and Jean van Heijenoort (Eds.). Vol. I. Clarendon Press ; Oxford University Press, Oxford [Oxfordshire] : New York.

John Harrison. 2009. HOL Light: An Overview. In *Theorem Proving in Higher Order Logics*, Stefan Berghofer, Tobias Nipkow, Christian Urban, and Makarius Wenzel (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 60–66. https://doi.org/10.1007/978-3-642-03359-9_4

Leon Henkin. 1950. Completeness in the theory of types. *Journal of Symbolic Logic* 15, 2 (1950), 8191. https://doi.org/10.2307/2266967

Martin Hofmann. 1995. *Extensional concepts in intensional type theory.* Ph.D. Dissertation. University of Edinburgh. http://www.lfcs.inf.ed.ac.uk/reports/95/ECS-LFCS-95-327/

Antonius J. C. Hurkens. 1995. A simplification of Girard's paradox. In *Typed Lambda Calculi and Applications*, Mariangiola Dezani-Ciancaglini and Gordon Plotkin (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 266–278. https://doi.org/10.1007/BFb0014058

Barry Jay. 2019. A Simpler Lambda Calculus. In *Proceedings of the 2019 ACM SIGPLAN Workshop on Partial Evaluation and Program Manipulation (PEPM 2019).* ACM, New York, NY, USA, 1–9. https://doi.org/10.1145/3294032.3294085

Ulrich Kohlenbach. 2008. *Systems based on classical logic and functional interpretation.* Springer Berlin Heidelberg, Berlin, Heidelberg, 163–197. https://doi.org/10.1007/978-3-540-77533-1_10

Ramana Kumar, Rob Arthan, Magnus O. Myreen, and Scott Owens. 2016. Self-Formalisation of Higher-Order Logic. *Journal of Automated Reasoning* 56, 3 (01 Mar 2016), 221–259. https://doi.org/10.1007/s10817-015-9357-x

Sigekatu Kuroda. 1951. Intuitionistische Untersuchungen der formalistischen Logik. *Nagoya Math. J.* 2 (1951), 35–47. https://projecteuclid.org:443/euclid.nmj/1118764737

Horst Luckhardt. 1973. *Elimination of extensionality.* Springer Berlin Heidelberg, Berlin, Heidelberg, 28–40. https://doi.org/10.1007/BFb0060874

Benjamin C Pierce. 2002. *Types and programming languages.* MIT Press, Cambridge, Mass.

Christine Rizkallah. 2009. *Proof Representations for Higher Order Logic.* Master's thesis. Universität des Saarlandes, Saarbrücken, Germany.

Morten Heine B. Sørensen and Pawel Urzyczyn. 1998. *Lectures on the Curry–Howard Isomorphism.* Elsevier Science.

Gaisi Takeuti. 1953. On a generalized logic calculus. *Japanese journal of mathematics: transactions and abstracts* 23 (1953), 39–96. https://doi.org/10.4099/jjm1924.23.0_39

Norbert Völker. 2007. HOL2P - A System of Classical Higher Order Logic with Second Order Polymorphism. In *Theorem Proving in Higher Order Logics*, Klaus Schneider and Jens Brandt (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 334–351. https://doi.org/10.1007/978-3-540-74591-4_25

Naim Çaman and J.Roger Hindley. 1998. Combinatory weak reduction in lambda calculus. *Theoretical Computer Science* 198, 1 (1998), 239 – 247. https://doi.org/10.1016/S0304-3975(97)00250-8

Appendix A

A.1 Polymorphic HOL

We defined two systems, extensional and intensional higher-order logic.

A.1.1 Extensional Polymorphic Higher-Order Logic

Syntax

Kinds

$$\begin{aligned} \kappa ::= {} & *_{\mathsf{S}} && \text{(small types)} \\ \mid {} & *_{\mathsf{L}} && \text{(large types)} \end{aligned}$$

Types

$$\begin{aligned} \tau ::= {} & o && \text{(propositions)} \\ \mid {} & \iota && \text{(individuals)} \\ \mid {} & \tau_1 \Rightarrow \tau_2 && \text{(functions)} \\ \mid {} & \alpha && \text{(type variables)} \\ \mid {} & \Pi\alpha.\ \tau && \text{(type polymorphism)} \end{aligned}$$

Terms

$$\begin{aligned} s, t, u, v ::= {} & x_\tau && \text{(variables)} \\ \mid {} & \lambda x_\tau.\ s && \text{(lambda abstraction)} \\ \mid {} & s\ t && \text{(application)} \\ \mid {} & [=]_\tau && \text{(equality)} \\ \mid {} & \Lambda\alpha.\ s && \text{(type abstraction)} \\ \mid {} & s\ [\!:\tau:\!] && \text{(type application)} \\ \mid {} & \varepsilon_\tau && \text{(indefinite description)} \end{aligned}$$

Where types are taken up to α equivalence, and terms are taken up to α and β equivalence.

Kinding Rules

$$\frac{}{\vdash o :_\kappa *_r} \qquad \frac{}{\vdash \iota :_\kappa *_r}$$

$$\frac{\vdash \tau_1 :_\kappa *_r \qquad \vdash \tau_2 :_\kappa *_r}{\vdash \tau_1 \Rightarrow \tau_2 :_\kappa *_r}$$

$$\frac{}{\vdash \alpha :_\kappa *_r} \qquad \frac{\vdash \tau :_\kappa *_r}{\vdash (\Pi\alpha.\ \tau) :_\kappa *_\mathsf{L}}$$

where r is either S or L.

Typing Rules

$$\frac{}{\vdash x_\tau : \tau}$$

$$\frac{\vdash s : \tau_2}{\vdash \lambda x_{\tau_1}.\ s : \tau_1 \Rightarrow \tau_2} \qquad \frac{\vdash s : \tau_1 \Rightarrow \tau_2 \qquad \vdash t : \tau_1}{\vdash s\ t : \tau_2}$$

$$\frac{\vdash s : \tau}{\vdash \Lambda\alpha.\ s : \Pi\alpha.\ \tau} \qquad \frac{\vdash s : \Pi\alpha.\ \tau \qquad \vdash \sigma :_\mathsf{R} \mathsf{S}}{\vdash s\ [\!:\sigma:\!] : \tau[\sigma/\alpha]}$$

$$\frac{}{\vdash [=]_\tau : \tau \Rightarrow \tau \Rightarrow o}$$

$$\frac{\vdash \tau :_\mathsf{R} r}{\vdash [\varepsilon]_\tau : (\tau \Rightarrow o) \Rightarrow \tau}$$

$$\frac{\vdash \tau :_\mathsf{R} r}{\vdash [\forall]_\tau : (\tau \Rightarrow o) \Rightarrow o} \qquad \frac{\vdash \tau :_\mathsf{R} r}{\vdash [\forall_\mathsf{ty}]_\tau : (\Pi\alpha.\ o) \Rightarrow o}$$

Proof Rules

Structural

$$\frac{}{p \vdash p}\ \text{assm}$$

$$\frac{\Gamma \vdash p}{\Gamma, \Delta \vdash p}\ \text{weaken} \qquad \frac{\Gamma, \Delta, \Delta \vdash p}{\Gamma, \Delta \vdash p}\ \text{contract}$$

Equality

$$\frac{}{\vdash s = s}\ \text{refl} \qquad \frac{\Gamma \vdash s = t \quad \Delta \vdash t = u}{\Gamma, \Delta \vdash s = u}\ \text{trans}$$

$$\frac{\Gamma \vdash s = t \quad \Delta \vdash u = v}{\Gamma, \Delta \vdash s\ u = t\ v}\ \text{mk-comb} \qquad \frac{}{\vdash (\lambda x.\ s)\ t = s[t/x]}\ \text{beta}$$

$$(x \text{ not in } \Gamma)\ \frac{\Gamma \vdash s = t}{\Gamma \vdash (\lambda x.\ s) = (\lambda x.\ t)}\ \text{abs} \qquad (x \text{ not in } t)\ \frac{}{\vdash (\lambda x.\ t\ x) = t}\ \text{eta}$$

$$\frac{\Gamma \vdash s = t}{\Gamma \vdash s\ [{:}\tau{:}] = t\ [{:}\tau{:}]}\ \text{tyapp} \qquad \frac{}{\vdash (\Lambda \alpha.\ s)\ [{:}\sigma{:}] = s[\sigma/\alpha]}\ \text{tybeta}$$

$$(\alpha \text{ not in } \Gamma)\ \frac{\Gamma \vdash s = t}{\Gamma \vdash (\Lambda \alpha.\ s) = (\Lambda \alpha.\ t)}\ \text{tyabs} \qquad (\alpha \text{ not in } f)\ \frac{}{\vdash (\Lambda \alpha.\ f\ [{:}\alpha{:}]) = f}\ \text{tyeta}$$

$$\frac{\Gamma \vdash s =_o t \quad \Delta \vdash s}{\Gamma, \Delta \vdash t}\ \text{eq-mp} \qquad \frac{\Gamma, q \vdash p \quad \Delta, p \vdash q}{\Gamma, \Delta \vdash p =_o q}\ \text{deduct-antisym}$$

Axioms

$$(x \text{ not in } p)\ \frac{}{\vdash p\ s \longrightarrow p\ (\varepsilon x.\ p\ x)}\ \text{select-ax} \qquad \frac{}{\vdash \exists (f : \iota \Rightarrow \iota).\ \text{inj}\ f \wedge \neg\, \text{onto}\ f}\ \text{infinity-ax}$$

A.1.2 Intensional Polymorphic Higher-Order Logic

Syntax

$$\begin{aligned} s, t, u, v ::= {} & \ldots \\ & \mid [\forall]_\tau && \text{(universal quantification)} \\ & \mid [\forall_{\mathsf{ty}}] && \text{(universal type-quantification)} \\ & \mid [\dot{=}]_\alpha && \text{(ext. equiv. on type-vars)} \end{aligned}$$

The syntax is as it is in $\text{PHOL}_{\xi\eta}$, except with the addition of $[\forall]_\tau$, $[\forall_{\mathsf{ty}}]$, and $[\dot{=}]$ as primitive symbols. The addition of the foralls is neccesary because the proofs of the introduction rules for these makes use of *abs* and *tyabs* respectively (which are not present in PHOL).

The addition of $[\dot{=}]_\alpha$ is necessary, because nothing in the logic reduces due to type-substition, except $[=]_\alpha$, which is too weak for the translation.

In addition, ***beta equivalence changes*** due to the removal of ***abs*** and ***tyabs***. The precise nature of this is that beta-reducts may only reduce if the reduct does not contain the bound variable of any lambda term enclosing the reduct. (See Subsection A.2.2 for a thorough definition.)

Typing

$$\frac{\vdash \tau :_\kappa k}{\vdash [\forall]_\tau : (\tau \Rightarrow o) \Rightarrow o} \qquad \frac{}{\vdash [\forall_{\mathsf{ty}}] : (\Pi\alpha.\ o) \Rightarrow o}$$

Rules

We remove ***eta***, ***abs***, ***tyeta*** and ***tyabs***. Moreover, we add the rules

$$(x \text{ not in } \Gamma)\ \frac{\Gamma \vdash p}{\Gamma \vdash \forall x.\, p}\ \forall\text{I} \qquad [s/x]\ \frac{\Gamma \vdash [\forall]\ p}{\Gamma \vdash p\ s}\ \forall\text{E}$$

$$(\alpha \text{ not in } \Gamma)\ \frac{\Gamma \vdash p}{\Gamma \vdash \forall_{\mathsf{ty}}\alpha.\ p}\ \forall_{\mathsf{ty}}\text{I} \qquad \frac{\Gamma \vdash [\forall_{\mathsf{ty}}]\ p}{\Gamma \vdash p\ [:\tau:]}\ \forall_{\mathsf{ty}}\text{E}$$

The definitions of the binder form of $\forall$ and $\forall_{\mathsf{ty}}$ are the same as in $\mathrm{PHOL}_{\xi\eta}$.

A.2 Rewriting Subterms

A.2.1 In $\mathrm{PHOL}_{\xi\eta}$

Define a ***context*** for $\mathrm{PHOL}_{\xi\eta}$ as

$$\begin{aligned} e ::= {} & [\cdot] \\ & \mid x \\ & \mid \lambda x.\ e \\ & \mid e_1\ e_2 \\ & \mid [=] \\ & \mid \Lambda\alpha.\ e \\ & \mid e\ [:\tau:] \\ & \mid \varepsilon \end{aligned}$$

A context represents a term with a *hole*, denoted by the symbol $[\cdot]$. Application of a term to a context is written $e[s]$, and represents the replacement of $[\cdot]$ by s, which results in a new term.

Using contexts, we can show the useful rule

$$\frac{\Gamma \vdash s = t \qquad \Delta \vdash e[s]}{\Gamma, \Delta \vdash e[t]} \text{ rewrite}$$

is admissible to PHOL.

The proof is by showing the lemma

$$\frac{\Gamma \vdash s = t}{\Gamma \vdash e[s] = e[t]}$$

is admissable by induction on e (the rewrite rule follows immediately by *eq-mp*).

Proof.
By induction on e

Case $[\cdot]$:

$$\frac{\Gamma \vdash s = t}{\Gamma \vdash ([\cdot])[s] = ([\cdot])[t]} \text{ defn-subst}$$

Case x: by *refl*.

Case $\lambda x.\ e$: by *abs*.

Case $e_1\ e_2$: by *mk-comb*.

Case $[=]$: by *refl*.

Case $\Lambda\alpha.\ e$: by *tyabs*.

Case $e\ [:\tau:]$: by *tyapp*.

Case ε: by *refl*. □

A.2.2 In PHOL

The major difference of PHOL, as compared to $\text{PHOL}_{\xi\eta}$, is that we do not have the *abs* or *tyabs* rules. The effect on the logic caused by the absence of these rules is that you cannot rewrite any subterm from under a lambda which contains the variable bound by that lambda. This is, in fact, just capture avoiding substitution, and thus we do not need contexts to show the admissibility of this rule.

The substition lemma for PHOL is

$$(x \text{ is free in } u)\ \frac{\Gamma \vdash s = t \qquad \Delta \vdash u[s/x]}{\Gamma, \Delta \vdash u[t/x]}\ \text{rewrite}$$

using the appropriate notion of context for PHOL. This is again proven from

$$\frac{\Gamma \vdash s = t}{\Gamma \vdash u[s/x] = u[s/x]}$$

by induction on u.

Proof.
By induction on u

Case y:

Subcase $y \equiv x$:

$$\frac{\Gamma \vdash s = t}{\Gamma \vdash x[s/x] = x[t/x]}\ \text{defn-subst}$$

Subcase $y \not\equiv x$: by *refl*.

Case $\lambda y.\ u$:
By the conditions on x, $y \not\equiv x$.

$$\frac{\dfrac{\dfrac{}{\vdash (\lambda z\ y.\ u[z/x]) = (\lambda z\ y.\ u[z/x])}\ \text{refl} \qquad \Gamma \vdash s = t}{\Gamma \vdash (\lambda z\ y.\ u[z/x])\ s = (\lambda z\ y.\ u[z/x])\ t}\ \text{mk-comb}}{\Gamma \vdash (\lambda y.\ e[s/x]) = (\lambda y.\ e[t/x])}\ \text{beta}$$

Case $u_1\ u_2$: by *mk-comb*.

Case $[=]$: by *refl*.

Case $\Lambda\alpha.\ u$:
If x_τ and τ contains a free α, we alpha-rename the type-lambdas fto avoid a variable clash.

$$\dfrac{\dfrac{\dfrac{}{\vdash (\lambda z.\ \Lambda\alpha.\ u[z/x]) = (\lambda z.\ \Lambda\alpha.\ u[z/x])}\ \text{refl} \qquad \Gamma \vdash s = t}{\Gamma \vdash (\lambda z.\ \Lambda\alpha.\ e[z/x])\ s = (\lambda z.\ \Lambda\alpha.\ e[z/x])\ t}\ \text{mk-comb}}{\Gamma \vdash (\Lambda\alpha.\ e[s/x]) = (\Lambda\alpha.\ e[t/x])}\ \text{beta}$$

Case $e\ [:\tau:]$: by *tyapp*.

Case ε: by *refl*. □

A.2.3 Of Assumptions

$$\dfrac{\Gamma \vdash s = t \qquad \Delta, e[s] \vdash q}{\Gamma, \Delta, e[t] \vdash q}\ \text{rewrite-asm}$$

Proof.
$$\dfrac{\dfrac{\Gamma \vdash s = t \qquad \dfrac{}{e[t] \vdash e[t]}\ \text{assm}}{\Gamma, e[t] \vdash e[s]}\ \text{rewrite} \qquad \Delta, e[s] \vdash q}{\Gamma, \Delta, e[t] \vdash q}\ \text{rewrite-asm}$$

□

A.3 Useful Rules

A.3.1 Assumption Substitution

$$\dfrac{\Gamma \vdash s \qquad s, \Delta \vdash t}{\Gamma, \Delta \vdash t}\ \text{ante-subst}$$

Firstly, note that this is not *cut* of the sequent calculus, as this is a natural deduction system.

Proof. This simple proof was pointed out to me by Johannes Åman Pohjola.

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash s}{\Gamma, t \vdash s}\ \text{weaken} \qquad s, \Delta \vdash t}{\Gamma, \Delta \vdash s =_o t}\ \text{deduct-antisym} \qquad \Gamma \vdash s}{\dfrac{\Gamma, \Delta, \Gamma \vdash t}{\Gamma, \Delta \vdash t}\ \text{contract}}\ \text{eq-mp}$$

□

A.3.2 Functional and Type-Functional Extensionality

In $\mathrm{PHOL}_{\xi\eta}$, the rule of functional extensionality

$$(\text{fresh } x)\ \frac{\Gamma \vdash f\ x = g\ x}{\Gamma \vdash f = g}\ \text{fun-ext} \qquad (\text{fresh } \alpha)\ \frac{\Gamma \vdash f\ [:\alpha:] = g\ [:\alpha:]}{\Gamma \vdash f = g}\ \text{ty-fun-ext}$$

is derivable as follows

$$\frac{(x \text{ not free in } f)\ \dfrac{}{\vdash (\lambda x.\ f\ x) = f}\ \text{eta} \quad (x \text{ not free in } g)\ \dfrac{}{\vdash (\lambda x.\ g\ x) = g}\ \text{eta} \quad \dfrac{\Gamma \vdash f\ x = g\ x}{\Gamma \vdash (\lambda x.\ f\ x) = (\lambda x.\ g\ x)}\ \text{abs}}{\Gamma \vdash f = g}\ \text{rewrite}^+$$

and

$$\frac{(\alpha \text{ not free in } f)\ \dfrac{}{\vdash (\Lambda \alpha.\ f\ [:\alpha:]) = f}\ \text{tyeta} \quad (\alpha \text{ not free in } g)\ \dfrac{}{\vdash (\Lambda x.\ g\ [:\alpha:]) = g}\ \text{tyeta} \quad \dfrac{\Gamma \vdash f\ x = g\ x}{\Gamma \vdash (\Lambda \alpha.\ f\ [:\alpha:]) = (\Lambda \alpha.\ g\ [:\alpha:])}\ \text{abs}}{\Gamma \vdash f = g}\ \text{rewrite}^+$$

A.3.3 Symmetry

Symmetry of equality

$$\frac{\Gamma \vdash s = t}{\Gamma \vdash t = s}\ \text{sym}$$

is derivable in PHOL as follows

$$\frac{\Gamma \vdash s =_\tau t \qquad \dfrac{}{\vdash s =_\tau s}\ \text{refl}}{\Gamma \vdash t =_\tau s}\ \text{rewrite}$$

A.4 Useful Definitions

A.4.1 The Connectives

The definitions of the connectives are as follows:

$$
\begin{aligned}
\top &\triangleq (\lambda p.\ p) = (\lambda p.\ p) \\
[\wedge] &\triangleq \lambda p\ q.\ (\lambda f.\ f\ p\ q) = (\lambda f.\ f\ \top\ \top) \\
p \wedge q &\triangleq [\wedge]\ p\ q \\
[\longrightarrow] &\triangleq \lambda p\ q.\ (p \wedge q) = p \\
p \longrightarrow q &\triangleq [\longrightarrow]\ p\ q \\
[\forall]_\tau &\triangleq \lambda p.\ p =_\tau (\lambda x.\ \top) \\
\forall x.\, p &\triangleq [\forall]\ (\lambda x.\ p) && \text{(where } p \text{ may contain } x\text{)} \\
[\exists] &\triangleq \lambda p.\ \forall z_o.(\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o \\
\exists x.\, p &\triangleq [\exists]\ (\lambda x.\ p) && \text{(where } p \text{ may contain } x\text{)} \\
[\vee] &\triangleq \lambda p\ q.\ \forall z_o.(p \longrightarrow z_o) \longrightarrow (q \longrightarrow z_o) \longrightarrow z_o \\
p \vee q &\triangleq [\vee]\ p\ q \\
\bot &\triangleq \forall z_o.\, z_o \\
[\neg] &\triangleq \lambda p.\ p \longrightarrow \bot \\
\neg p &\triangleq [\neg]\ p \\
[\forall_{\mathsf{ty}}] &\triangleq \lambda z.\ z =_{\Pi\alpha.\ o} (\Lambda\alpha.\ \top) \\
\forall_{\mathsf{ty}}\alpha.\ p &\triangleq [\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p) && \text{(where } p \text{ may contain } \alpha\text{)}
\end{aligned}
$$

$$
\begin{aligned}
\text{inj} &\triangleq \lambda f.\ \forall x\ y.\, f\ x = f\ y \longrightarrow x = y \\
\text{onto} &\triangleq \lambda f.\ \forall y.\, \exists x.\, f\ x = y
\end{aligned}
$$

Their introduction and elimination rules are

$$\frac{}{\vdash \top}\ \top\text{I} \qquad \frac{\Gamma \vdash p}{\Gamma, \top \vdash p}\ \top\text{E}$$

$$\frac{\Gamma \vdash p \qquad \Delta \vdash q}{\Gamma, \Delta \vdash p \wedge q}\ \wedge\text{I} \qquad \frac{\Gamma \vdash p \wedge q}{\Gamma \vdash p}\ \wedge\text{E-L} \qquad \frac{\Gamma \vdash p \wedge q}{\Gamma \vdash q}\ \wedge\text{E-R}$$

$$\frac{\Gamma, p \vdash q}{\Gamma \vdash p \longrightarrow q}\ \rightarrow\text{I} \qquad \frac{\Gamma \vdash p \longrightarrow q \qquad \Delta \vdash p}{\Gamma, \Delta \vdash q}\ \rightarrow\text{E}$$

$$(x \text{ not in } \Gamma)\ \frac{\Gamma \vdash p}{\Gamma \vdash \forall x.\, p}\ \forall\text{I} \qquad \frac{\Gamma \vdash [\forall]\ p}{\Gamma \vdash p\ s}\ \forall\text{E}$$

$$\frac{\Gamma \vdash p\ s}{\Gamma \vdash [\exists]\ p}\ \exists\text{I} \qquad (x \text{ not in } \Gamma)\ \frac{\Gamma \vdash \exists x.\, p}{\Gamma \vdash p}\ \exists\text{E}$$

$$\frac{\Gamma \vdash p}{\Gamma \vdash p \vee q}\ \vee\text{I-L} \qquad \frac{\Gamma \vdash q}{\Gamma \vdash p \vee q}\ \vee\text{I-R} \qquad \frac{\Gamma \vdash p_0 \vee p_1 \qquad \Delta, p_0 \vdash q \qquad \Delta, p_1 \vdash q}{\Gamma, \Delta \vdash q}\ \vee\text{E}$$

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash p}\ \bot\text{E}$$

$$\frac{\Gamma, p \vdash \bot}{\Gamma \vdash \neg p}\ \neg\text{I} \qquad \frac{\Gamma \vdash \neg p \qquad \Delta \vdash p}{\Gamma, \Delta \vdash q}\ \neg\text{E}$$

$$(\alpha \text{ not in } \Gamma)\ \frac{\Gamma \vdash p}{\Gamma \vdash \forall_{\mathsf{ty}} \alpha.\ p}\ \forall_{\mathsf{ty}}\text{I} \qquad \frac{\Gamma \vdash [\forall_{\mathsf{ty}}]\ p}{\Gamma \vdash p\ [\!:\tau:\!]}\ \forall_{\mathsf{ty}}\text{E}$$

Truth

Introduction

$$\dfrac{\dfrac{}{\vdash (\lambda p.\ p) = (\lambda p.\ p)}\ \text{refl}}{\vdash \top}\ \text{defn}\top$$

Elimination

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\top \vdash \top}\ \text{assm}}{\top \vdash (\lambda q.\ q) = (\lambda q.\ q)}\ \text{defn} \qquad \dfrac{}{\vdash p = p}\ \text{refl}}{\top \vdash (\lambda q.\ q)\ p = (\lambda q.\ q)\ p}\ \text{mk-conv}}{\top \vdash p = p}\ \text{beta} \qquad \dfrac{}{\Gamma, \top \vdash p}\ \Gamma \vdash p\ \text{eq-mp}$$

Conjunction

Introduction

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \top}}{p \vdash \top}\,\text{weaken} \quad \dfrac{\Gamma \vdash p}{\Gamma, \top \vdash p}\,\top\text{E}}{\Gamma \vdash \top = p}\,\text{deduct-antisym} \quad \dfrac{\dfrac{\dfrac{}{\vdash \top}\,\top\text{I}}{q \vdash \top}\,\text{weaken} \quad \dfrac{\Delta \vdash q}{\Delta, \top \vdash q}\,\top\text{E}}{\Delta \vdash \top = q}\,\text{deduct-antisym} \quad \dfrac{}{\vdash (\lambda f.\ f\ \top\ \top) = (\lambda f.\ f\ \top\ \top)}\,\text{refl}}{\Gamma, \Delta \vdash p \wedge q}\,\text{defn}\wedge + \text{beta}$$

Elimination

The left rule has the proof

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash p \wedge q}{\Gamma \vdash (\lambda f.\ f\ p\ q) = (\lambda f.\ f\ \top\ \top)}\,\text{defn} \quad \dfrac{}{\vdash (\lambda x\ y.\ x) = (\lambda x\ y.\ x)}\,\text{refl}}{\Gamma \vdash (\lambda f.\ f\ p\ q)\ (\lambda x\ y.\ x) =_o (\lambda f.\ f\ \top\ \top)\ (\lambda x\ y.\ x)}\,\text{mk-comb}}{\Gamma \vdash (\lambda x\ y.\ x)\ p\ q =_o (\lambda x\ y.\ x)\ \top\ \top}\,\text{beta}}{\Gamma \vdash p =_o \top}\,\text{beta}}{\Gamma \vdash \top =_o p}\,\text{sym} \quad \dfrac{}{\vdash \top}\,\top\text{I}}{\Gamma \vdash p}\,\text{eq-mp}$$

and similarly for the right rule.

Implication

Introduction

$$\dfrac{\dfrac{\dfrac{\dfrac{}{p \vdash p}\,\text{assm} \quad \dfrac{\Gamma, p \vdash q}{\Gamma, p, p \vdash p \wedge q}\,\wedge\text{I}}{\Gamma, p \vdash p \wedge q}\,\text{contract} \quad \dfrac{\dfrac{}{p \wedge q \vdash p \wedge q}\,\text{assm}}{p \wedge q \vdash p}\,\wedge\text{E-1}}{\Gamma \vdash (p \wedge q) =_o p}\,\text{deduct-antisym}}{\Gamma \vdash p \longrightarrow q}\,\text{defn}\rightarrow$$

Elimination

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash p \longrightarrow q}{\Gamma \vdash (p \wedge q) =_o p}\ \text{defn}\!\rightarrow}{\Gamma \vdash p =_o (p \wedge q)}\ \text{sym} \qquad \Delta \vdash p}{\dfrac{\Gamma, \Delta \vdash p \wedge q}{\Gamma, \Delta \vdash q}\ \wedge\text{E-2}}\ \text{eq-mp}$$

Universal Quantification

N.B. in PHOL, the $\forall$ symbol and its introduction and elimination rules are axiomatic.

Introduction

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash p}{\Gamma, \top \vdash p}\ \top\text{E} \qquad \dfrac{\dfrac{}{\vdash \top}\ \top\text{I}}{p \vdash \top}\ \text{weaken}}{\Gamma \vdash p =_o \top}\ \text{deduct-antisymm}}{(x \text{ fresh})\ \Gamma \vdash (\lambda x.\ p) =_{\tau \Rightarrow o} (\lambda x.\ \top)}\ \text{abs}}{\Gamma \vdash (\lambda z.\ z =_{\tau \Rightarrow o} (\lambda x.\ \top))\ (\lambda x.\ p)}\ \text{beta}}{\Gamma \vdash \forall x.\, p}\ \text{defn}\forall$$

Elimination

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash [\forall]\ p}{\Gamma \vdash p =_{\tau \Rightarrow o} (\lambda x.\ \top)}\ \text{defn}\forall + \text{beta}}{\Gamma \vdash (\lambda x.\ \top) =_{\tau \Rightarrow o} p}\ \text{sym} \qquad \dfrac{\dfrac{}{\vdash \top}\ \top\text{I}}{\vdash (\lambda x.\ \top)\ s}\ \text{beta}}{\Gamma \vdash p\ s}\ \text{rewrite}$$

Existential Quantification

Introduction

$$\dfrac{\dfrac{\dfrac{}{\forall x.\, p\ x \longrightarrow q \vdash \forall x.\, p\ x \longrightarrow q}\ \text{assm}}{\forall x.\, p \longrightarrow q \vdash p\ s \longrightarrow q}\ \forall\text{E + beta} \qquad \Gamma \vdash p\ s}{\dfrac{\Gamma, \forall x.\, p\ x \longrightarrow q \vdash q}{\Gamma \vdash (\forall x.\, p\ x \longrightarrow q) \longrightarrow q}\ \to\text{I}}\ \to\text{E}$$
$$[s/x] \qquad (q \text{ fresh})\ \dfrac{\Gamma \vdash (\forall x.\, p\ x \longrightarrow q) \longrightarrow q}{\Gamma \vdash \forall q.\, (\forall x.\, p\ x \longrightarrow q) \longrightarrow q}\ \forall\text{I}$$
$$\dfrac{\Gamma \vdash \forall q.\, (\forall x.\, p\ x \longrightarrow q) \longrightarrow q}{\Gamma \vdash [\exists]\ p}\ \text{defn}\exists$$

Elimination

$$\dfrac{\dfrac{\Gamma \vdash \exists x.\, p}{\Gamma \vdash \forall q.\, (\forall x.\, p \longrightarrow q) \longrightarrow q}\ \text{defn}\exists \qquad \dfrac{\dfrac{\dfrac{}{p \vdash p}\ \text{assm}}{\vdash p \longrightarrow p}\ \to\text{I}}{\vdash \forall x.\, p \longrightarrow p}\ \forall\text{I}}{\Gamma \vdash (\forall x.\, p \longrightarrow p) \longrightarrow p}\ \forall\text{E} \qquad (x \text{ fresh})$$
$$[p/q] \qquad \dfrac{}{\Gamma \vdash p}\ \to\text{E}$$

Disjunction

Introduction

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{p \longrightarrow r \vdash p \longrightarrow r}\ \text{assm}}{p \longrightarrow r, q \longrightarrow r \vdash p \longrightarrow r}\ \text{weaken} \qquad \Gamma \vdash p}{\Gamma, p \longrightarrow r, q \longrightarrow r \vdash r}\ \to\text{E}}{\Gamma \vdash (p \longrightarrow r) \longrightarrow (q \longrightarrow r) \longrightarrow r}\ \to\text{I}^{+}}{\Gamma \vdash \forall r.\, (p \longrightarrow r) \longrightarrow (q \longrightarrow r) \longrightarrow r}\ \vee\text{I-L}}{\Gamma \vdash p \vee q}\ \text{defn}\vee \qquad (\text{fresh } r)$$

And similarly for $\vee$I-2.

Elimination

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash p \vee q}{\Gamma \vdash \forall r.\, (p \longrightarrow r) \longrightarrow (q \longrightarrow r) \longrightarrow r}\ \text{defn}\vee}{\Gamma \vdash (p \longrightarrow r) \longrightarrow (q \longrightarrow r) \longrightarrow r}\ \forall\text{E} \qquad \dfrac{\Delta, p \vdash r}{\Delta \vdash p \longrightarrow r}\ \to\text{I}}{\Gamma, \Delta \vdash (q \longrightarrow r) \longrightarrow r}\ \to\text{E} \qquad \dfrac{\Delta, q \vdash r}{\Delta \vdash q \longrightarrow r}\ \to\text{I}}{\dfrac{\Gamma, \Delta, \Delta \vdash r}{\Gamma, \Delta \vdash r}\ \text{contract}}\ \to\text{E}$$
$$[r/r]$$

False

Elimination

$$[p/p]\ \dfrac{\dfrac{\Gamma \vdash \bot}{\Gamma \vdash \forall p.\, p}\ \text{defn}\bot}{\Gamma \vdash p}\ \forall\text{E}$$

Not

Introduction

$$\dfrac{\dfrac{\Gamma, p \vdash \bot}{\Gamma \vdash p \longrightarrow \bot}\ \rightarrow\text{I}}{\Gamma \vdash \neg p}\ \text{defn}\neg$$

Elimination

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash \neg p}{\Gamma \vdash p \longrightarrow \bot}\ \text{defn}\neg \qquad \Delta \vdash p}{\Gamma, \Delta \vdash \bot}\ \rightarrow\text{E}}{\Gamma, \Delta \vdash q}\ \bot\text{E}$$

Type Forall

N.B. in PHOL, the $\forall_{\mathsf{ty}}$ symbol and its introduction and elimination rules are axiomatic.

Introduction

$$(\text{fresh } \alpha)\ \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \top}\ \top\text{I}}{p \vdash \top}\ \text{weaken} \qquad \dfrac{\Gamma \vdash p}{\Gamma, \top \vdash p}\ \top\text{E}}{\Gamma \vdash p = \top}\ \text{deduct-antisym}}{\Gamma \vdash (\Lambda\alpha.\, p) =_{\Pi\alpha.\, o} (\Lambda\alpha.\, \top)}\ \text{tyabs}}{\Gamma \vdash (\lambda z.\, z =_{\Pi\alpha.\, o} (\Lambda\alpha.\, \top))\ (\Lambda\alpha.\, p)}\ \text{tybeta}}{\Gamma \vdash \forall_{\mathsf{ty}}\alpha.\ p}\ \text{defn}\forall_{\mathsf{ty}}$$

Elimination

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash [\forall_{\mathsf{ty}}]\ s}{\Gamma \vdash s = (\Lambda\alpha.\ \top)}\ \text{defn}\forall_{\mathsf{ty}} + \text{beta}}{\Gamma \vdash (\Lambda\alpha.\ \top) = s}\ \text{sym} \qquad \dfrac{\dfrac{\dfrac{}{\vdash \top}\ \top\text{I}}{\vdash \top[\tau/\alpha]}\ \text{defn-subst}}{\vdash (\Lambda\alpha.\ \top)\ [\!:\tau:\!]}\ \text{tybeta}}{\Gamma \vdash s\ [\!:\tau:\!]}\ \text{rewrite}$$

A.4.2 Other Useful Results About the Connectives

trans$\rightarrow$

$$\dfrac{\Gamma \vdash p \longrightarrow q \qquad \Delta \vdash q \longrightarrow r}{\Gamma, \Delta \vdash p \longrightarrow r}$$

$$\dfrac{\dfrac{\Delta \vdash q \longrightarrow r \qquad \dfrac{\Gamma \vdash p \longrightarrow q \qquad \dfrac{}{p \vdash p}\ \text{assm}}{\Gamma, p \vdash q}\ \rightarrow\text{E}}{\Gamma, \Delta, p \vdash r}\ \rightarrow\text{E}}{\Gamma, \Delta \vdash p \longrightarrow r}\ \rightarrow\text{I}$$

A.4.3 inj **and** onto

The usual definitions for inj and onto hold.

$$\begin{array}{lcl} \text{inj}\ f & \triangleq & \forall x\ y.\ f\ x = f\ y \longrightarrow x = y \\ \text{onto}\ f & \triangleq & \forall y.\ \exists x.\ f\ x = y \end{array}$$

A.5 Admissibility of inst and ty-inst

Structural

$$\frac{}{p \vdash p}\ \text{assm}$$

$$\frac{\Gamma \vdash p}{\Gamma, \Delta \vdash p}\ \text{weaken} \qquad \frac{\Gamma, \Delta, \Delta \vdash p}{\Gamma, \Delta \vdash p}\ \text{contract}$$

Equality

$$\frac{}{\vdash s = s}\ \text{refl} \qquad \frac{\Gamma \vdash s = t \quad \Delta \vdash t = u}{\Gamma, \Delta \vdash s = u}\ \text{trans}$$

$$\frac{\Gamma \vdash s = t \quad \Delta \vdash u = v}{\Gamma, \Delta \vdash s\ u = t\ v}\ \text{mk-comb}$$

$$\frac{\Gamma \vdash s = t}{\Gamma \vdash s\ [:\tau:] = t\ [:\tau:]}\ \text{tyapp}$$

$$\frac{\Gamma \vdash s =_o t \quad \Delta \vdash s}{\Gamma, \Delta \vdash t}\ \text{eq-mp} \qquad \frac{\Gamma, q \vdash p \quad \Delta, p \vdash q}{\Gamma, \Delta \vdash p =_o q}\ \text{deduct-antisym}$$

Axioms

$$(x \text{ not in } p)\ \frac{}{\vdash p\ s \longrightarrow p; (\varepsilon x.\, p\ x)}\ \text{select-ax} \qquad \frac{}{\vdash \exists (f : \iota \Rightarrow \iota).\, \text{inj}\ f \wedge \neg\, \text{onto}\ f}\ \text{infinity-ax}$$

Extensional Rules

$$(x \text{ not in } \Gamma)\ \frac{\Gamma \vdash s = t}{\Gamma \vdash (\lambda x.\ s) = (\lambda x.\ t)}\ \text{abs} \qquad (x \text{ not in } t)\ \frac{}{\vdash (\lambda x.\ t\ x) = t}\ \text{eta}$$

$$(\alpha \text{ not in } \Gamma)\ \frac{\Gamma \vdash s = t}{\Gamma \vdash (\Lambda \alpha.\ s) = (\Lambda \alpha.\ t)}\ \text{tyabs} \qquad (\alpha \text{ not in } f)\ \frac{}{\vdash (\Lambda \alpha.\ f\ [:\alpha:]) = f}\ \text{tyeta}$$

Intensional Rules

$$(x \text{ not in } \Gamma)\ \frac{\Gamma \vdash p}{\Gamma \vdash \forall x.\, p}\ \forall \text{I} \qquad \frac{\Gamma \vdash [\forall]\ p}{\Gamma \vdash p\ s}\ \forall \text{E}$$

$$(\alpha \text{ not in } \Gamma)\ \frac{\Gamma \vdash p}{\Gamma \vdash \forall_{\text{ty}} \alpha.\ p}\ \forall_{\text{ty}} \text{I} \qquad \frac{\Gamma \vdash [\forall_{\text{ty}}]\ p}{\Gamma \vdash p\ [:\tau:]}\ \forall_{\text{ty}} \text{E}$$

A.5.1 inst

The rule *inst* is *admissible* to $\text{PHOL}_{\xi\eta}$ and PHOL.

$$\frac{\Gamma \vdash s}{\Gamma[\theta] \vdash s[\theta]}\ \text{inst}$$

Proof. Induction on the proof-tree of $\Gamma \vdash s$.

Note that in these proofs, if a substitution occurs under a lambda (e.g. $(\lambda x.\ s[\theta])$), we assume that x does not occur in the free-variables of any term resulting from the substitution, as any lambda-term in which this *does* happen can be alpha-renamed such that it doesn't. We also make this assumption for type-lambdas.

Case assm:

$$\dfrac{\dfrac{}{p \vdash p}\ \text{assm}}{p[\theta] \vdash p[\theta]}\ \text{inst} \qquad \rightsquigarrow \qquad \dfrac{}{p[\theta] \vdash p[\theta]}\ \text{assm}$$

Case weaken:

$$\dfrac{\dfrac{\Gamma \vdash p}{\Gamma, \Delta \vdash p}\ \text{weaken}}{\Gamma[\theta], \Delta[\theta] \vdash p[\theta]}\ \text{inst} \qquad \rightsquigarrow \qquad \dfrac{\dfrac{\Gamma \vdash p}{\Gamma[\theta] \vdash p[\theta]}\ \text{I.H.}}{\Gamma[\theta], \Delta[\theta] \vdash p[\theta]}\ \text{weaken}$$

Case contract:

$$\dfrac{\dfrac{\Gamma, \Delta, \Delta \vdash p}{\Gamma, \Delta \vdash p}\ \text{contract}}{\Gamma[\theta], \Delta[\theta] \vdash p[\theta]}\ \text{inst} \qquad \rightsquigarrow \qquad \dfrac{\dfrac{\Gamma, \Delta, \Delta \vdash p}{\Gamma[\theta], \Delta[\theta], \Delta[\theta] \vdash p[\theta]}\ \text{I.H.}}{\Gamma[\theta], \Delta[\theta] \vdash p[\theta]}\ \text{contract}$$

Case refl:

$$\dfrac{\dfrac{}{\vdash s =_\tau s}\ \text{refl}}{\vdash (s =_\tau s)[\theta]}\ \text{inst} \qquad \rightsquigarrow \qquad \dfrac{\dfrac{}{\vdash s[\theta] =_\tau s[\theta]}\ \text{refl}}{\vdash (s =_\tau s)[\theta]}\ \text{defn-subst}$$

Case trans:

$$\dfrac{\dfrac{\Gamma \vdash s =_\tau t \qquad \Delta \vdash t =_\tau u}{\Gamma, \Delta \vdash s =_\tau u}\ \text{trans}}{\Gamma[\theta], \Delta[\theta] \vdash (s =_\tau u)[\theta]}\ \text{inst} \qquad \rightsquigarrow \qquad \dfrac{\dfrac{\dfrac{\Gamma \vdash s =_\tau t}{\Gamma[\theta] \vdash (s =_\tau t)[\theta]}\ \text{I.H.}}{\Gamma[\theta] \vdash s[\theta] =_\tau t[\theta]}\ \text{defn-subst} \qquad \dfrac{\dfrac{\Delta \vdash t =_\tau u}{\Delta[\theta] \vdash (t =_\tau u)[\theta]}\ \text{I.H.}}{\Delta[\theta] \vdash t[\theta] =_\tau u[\theta]}\ \text{defn-subst}}{\Gamma[\theta], \Delta[\theta] \vdash s[\theta] =_\tau u[\theta]}\ \text{trans}}{\Gamma[\theta], \Delta[\theta] \vdash (s =_\tau u)[\theta]}\ \text{defn-subst}$$

Case mk-comb:

$$\dfrac{\dfrac{\Gamma \vdash s = t \qquad \Delta \vdash u = v}{\Gamma, \Delta \vdash s\ u =_{\tau_2} t\ v}\ \text{mk-comb}}{\Gamma[\theta], \Delta[\theta] \vdash (s\ u =_{\tau_2} t\ v)[\theta]}\ \text{I.H.} \qquad \rightsquigarrow \qquad \dfrac{\dfrac{\dfrac{\Gamma \vdash s = t}{\Gamma[\theta] \vdash (s =_{\tau_1 \Rightarrow \tau_2} t)[\theta]}\ \text{I.H.}}{\Gamma[\theta] \vdash s[\theta] =_{\tau_1 \Rightarrow \tau_2} t[\theta]}\ \text{defn-subst} \qquad \dfrac{\dfrac{\Delta \vdash u = v}{\Gamma[\theta], \Delta[\theta] \vdash (u =_{\tau_1} v)[\theta]}\ \text{I.H.}}{\Gamma[\theta], \Delta[\theta] \vdash u[\theta] =_{\tau_2} v[\theta]}\ \text{defn-subst}}{\Gamma[\theta], \Delta[\theta] \vdash s[\theta]\ u[\theta] =_{\tau_2} t[\theta]\ v[\theta]}\ \text{mk-comb}}{\Gamma[\theta], \Delta[\theta] \vdash (s\ u =_{\tau_2} t\ v)[\theta]}\ \text{defn-subst}$$

Case beta:

$$\dfrac{\dfrac{}{\vdash (\lambda x.\ s)\ t = s[t/x]}\ \text{beta}}{\vdash \big((\lambda x.\ s)\ t = s[t/x]\big)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{}{\vdash (\lambda x.\ s[\theta - x])\ t[\theta] = (s[\theta - x])[t[\theta]/x]}\ \text{beta}}{\vdash \big((\lambda x.\ s)\ t\big)[\theta] = s[\theta - x, t[\theta]/x]}\ \text{defn-subst}}{\vdash \big((\lambda x.\ s)\ t\big)[\theta] = s[t/x, \theta]}\ \text{defn-subst}}{\vdash \big((\lambda x.\ s)\ t = s[t/x]\big)[\theta]}\ \text{defn-subst}$$

Case abs:

$$(x_\sigma \text{ not in } \Gamma)\ \dfrac{\dfrac{\Gamma \vdash s = t}{\Gamma \vdash (\lambda x_\sigma.\ s) =_\tau (\lambda x_\sigma.\ t)}\ \text{abs}}{\Gamma[\theta] \vdash \big((\lambda x_\sigma.\ s) =_\tau (\lambda x_\sigma.\ t)\big)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad (x_\sigma \text{ not in } \Gamma)\ \dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash s = t}{\Gamma[\theta - x] \vdash (s =_\tau t)[\theta - x]}\ \text{I.H.}}{\Gamma[\theta - x] \vdash s[\theta - x] =_\tau t[\theta - x]}\ \text{defn-subst}}{\Gamma[\theta] \vdash s[\theta - x] =_\tau t[\theta - x]}\ \text{defn-subst}}{\Gamma[\theta] \vdash (\lambda x.\ s[\theta - x]) =_\tau (\lambda x.\ t[\theta - x])}\ \text{abs}}{\Gamma[\theta] \vdash \big((\lambda x_\sigma.\ s) =_\tau (\lambda x_\sigma.\ t)\big)[\theta]}\ \text{defn-subst}$$

Case eta:

$$(x_\sigma \text{ not in } t)\ \dfrac{\dfrac{}{\vdash (\lambda x.\ t\ x_\sigma) =_{\sigma \Rightarrow \tau} t}\ \text{eta}}{\vdash \big((\lambda x.\ t\ x) =_{\sigma \Rightarrow \tau} t\big)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad (x \text{ not in } t[\theta - x])\ \dfrac{\dfrac{}{\vdash (\lambda x.\ t[\theta - x]\ x) =_{\sigma \Rightarrow \tau} t[\theta - x]}\ \text{eta}}{\vdash \big((\lambda x.\ t\ x) =_{\sigma \Rightarrow \tau} t\big)[\theta]}\ \text{defn-subst}$$

Case tyapp:

$$\dfrac{\dfrac{\Gamma \vdash s =_{\Pi\alpha.\ \tau} t}{\Gamma \vdash s\ [:\sigma:] =_\tau t\ [:\sigma:]}\ \text{tyapp}}{\Gamma[\theta] \vdash (s\ [:\sigma:] =_\tau t\ [:\sigma:])[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\Gamma \vdash s =_{\Pi\alpha.\ \tau} t}{\Gamma[\theta] \vdash (s =_{\Pi\alpha.\ \tau} t)[\theta]}\ \text{I.H.}}{\Gamma[\theta] \vdash s[\theta] =_{\Pi\alpha.\ \tau} t[\theta]}\ \text{defn-subst}}{\Gamma[\theta] \vdash s[\theta]\ [:\sigma:] =_\tau t[\theta]\ [:\sigma:]}\ \text{tyapp}}{\Gamma[\theta] \vdash \big(s\ [:\sigma:] =_\tau t\ [:\sigma:]\big)[\theta]}\ \text{defn-subst}$$

Case tybeta:

$$\dfrac{\dfrac{}{\vdash (\Lambda\alpha.\ s)\ [:\sigma:] = s[\sigma/\alpha]}\ \text{tybeta}}{\vdash \big((\Lambda\alpha.\ s)\ [:\sigma:] = s[\sigma/\alpha]\big)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{}{\vdash (\Lambda\alpha.\ s[\theta[\sigma/\alpha]])\ [:\sigma:] = (s[\theta[\sigma/\alpha]])[\sigma/\alpha]}\ \text{tybeta}}{\vdash ((\Lambda\alpha.\ s)\ [:\sigma:])[\theta] = (s[\sigma/\alpha])[\theta]}\ \text{defn-subst}$$

Case tyabs:

$$(\alpha \text{ not in } \Gamma)\ \dfrac{\dfrac{\Gamma \vdash s =_\tau t}{\vdash (\Lambda\alpha.\ s) =_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)}\ \text{tyabs}}{\Gamma[\theta] \vdash \big((\Lambda\alpha.\ s) =_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)\big)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad (\alpha \text{ not in } \Gamma[\theta])\ \dfrac{\dfrac{\dfrac{\Gamma \vdash s =_\tau t}{\Gamma[\theta] \vdash (s =_\tau t)[\theta]}\ \text{I.H.}}{\Gamma[\theta] \vdash s[\theta] =_\tau t[\theta]}\ \text{defn-subst}}{\Gamma[\theta] \vdash (\Lambda\alpha.\ s[\theta]) =_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t[\theta])}\ \text{tyabs}}{\Gamma[\theta] \vdash \big((\Lambda\alpha.\ s) =_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)\big)[\theta]}\ \text{defn-subst}$$

Case tyeta:

$$(\alpha \text{ not in } t)\ \dfrac{\dfrac{}{\vdash (\Lambda\alpha.\ t\ [:\alpha:]) =_{\Pi\alpha.\ \tau} t}\ \text{eta}}{\vdash \big((\Lambda\alpha.\ t\ [:\alpha:]) =_{\Pi\alpha.\ \tau} t\big)[\theta]}\ \text{I.H.} \quad \rightsquigarrow \quad \begin{array}{l}(\alpha \text{ not in } t[\theta])\\ (\alpha \text{ not in } t)\end{array}\ \dfrac{\dfrac{\dfrac{}{\vdash (\Lambda\alpha.\ t[\theta]\ [:\alpha:]) =_{\Pi\alpha.\ \tau} t[\theta]}\ \text{eta}}{\vdash (\Lambda\alpha.\ t[\theta]\ [:\alpha:]) =_{\Pi\alpha.\ \tau} t[\theta]}\ \text{defn-subst}}{\vdash \big((\Lambda\alpha.\ t\ [:\alpha:]) =_{\Pi\alpha.\ \tau} t\big)[\theta]}\ \text{defn-subst}$$

Case eq-mp:

$$\dfrac{\dfrac{\Gamma \vdash s =_o t \qquad \Delta \vdash s}{\Gamma, \Delta \vdash t}\ \text{eq-mp}}{\Gamma[\theta], \Delta[\theta] \vdash t[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\Gamma \vdash s =_o t}{\Gamma[\theta] \vdash (s =_o t)[\theta]}\ \text{I.H.}}{\Gamma[\theta] \vdash s[\theta] =_o t[\theta]}\ \text{defn-subst} \qquad \dfrac{\Delta \vdash s}{\Delta[\theta] \vdash s[\theta]}\ \text{I.H.}}{\Gamma[\theta], \Delta[\theta] \vdash t[\theta]}\ \text{eq-mp}$$

Case deduct-antisym:

$$\dfrac{\dfrac{\Gamma, q \vdash p \qquad \Delta, p \vdash q}{\Gamma, \Delta \vdash p =_o q}\ \text{deduct-antisym}}{\Gamma[\theta], \Delta[\theta], \vdash (p =_o q)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\Gamma, q \vdash p}{\Gamma[\theta], q[\theta] \vdash p[\theta]}\ \text{I.H.} \qquad \dfrac{\Delta \vdash q}{\Delta[\theta], p[\theta] \vdash q[\theta]}\ \text{I.H.}}{\Gamma[\theta], \Delta[\theta] \vdash p[\theta] =_o q[\theta]}\ \text{deduct-antisym}}{\Gamma[\theta], \Delta[\theta] \vdash (p =_o q)[\theta]}\ \text{defn-subst}$$

Case select-ax:

$$\dfrac{(x \text{ not in } p)\ \dfrac{}{\vdash p\ s \longrightarrow p\ (\varepsilon x_\tau.\, p\ x_\tau)}\ \text{select-ax}}{\vdash (p\ s \longrightarrow p\ (\varepsilon x_\tau.\, p\ x_\tau))[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{(x \text{ not in } p[\theta - x])\ \dfrac{}{\vdash p[\theta - x]\ s[\theta] \longrightarrow p[\theta - x]\ (\varepsilon x_\tau.\, p[\theta - x]\ x)}\ \text{select-ax}}{(x \text{ not in } p[\theta - x])\ \dfrac{\vdash p[\theta - x]\ s[\theta] \longrightarrow (p\ (\varepsilon x_\tau.\, p\ x))[\theta - x]}{\vdash (p\ x \longrightarrow p\ (\varepsilon x.\, p\ x))[\theta]}\ \text{defn-subst}}\ \text{defn-subst}$$

Case infinity-ax:

$$\dfrac{\dfrac{}{\vdash \exists (f : \iota \Rightarrow \iota).\, \text{inj}\ f \wedge \neg\, \text{onto}\ f}\ \text{infinity-ax}}{\vdash \big(\exists (f : \iota \Rightarrow \iota).\, \text{inj}\ f \wedge \neg\, \text{onto}\ f\big)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{}{\vdash \exists (f : \iota \Rightarrow \iota).\, \text{inj}\ f \wedge \neg\, \text{onto}\ f}\ \text{infinity-ax}}{\vdash \big(\exists (f : \iota \Rightarrow \iota).\, \text{inj}\ f \wedge \neg\, \text{onto}\ f\big)[\theta]}\ \text{defn-subst}$$

Case $\forall$I:

$$\dfrac{(x \text{ not in } \Gamma)\ \dfrac{\Gamma \vdash p}{\Gamma \vdash \forall x.\, p}\ \forall\text{I}}{\Gamma[\theta] \vdash (\forall x.\, p)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{(x \text{ not in } \Gamma[\theta - x])\ \dfrac{\dfrac{\Gamma \vdash p}{\Gamma[\theta - x] \vdash p[\theta - x]}\ \text{I.H.}}{\Gamma[\theta] \vdash \forall x.\, p[\theta - x]}\ \forall\text{I}}{(x \text{ not in } \Gamma)\ \Gamma[\theta] \vdash (\forall x.\, p)[\theta]}\ \text{defn-subst}$$

Case $\forall$E:

$$\dfrac{\dfrac{\Gamma \vdash [\forall]\ p}{\Gamma \vdash p\ s}\ \forall\text{E}}{\Gamma[\theta] \vdash (p\ s)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash [\forall]\ p}{\Gamma[\theta] \vdash ([\forall]\ p)[\theta]}\ \text{I.H.}}{\Gamma[\theta] \vdash [\forall]\ p[\theta]}\ \text{defn-subst}}{\Gamma[\theta] \vdash p[\theta]\ s[\theta]}\ \forall\text{E}}{\Gamma[\theta] \vdash (p\ s)[\theta]}\ \text{defn-subst}$$

Case $\forall_{\text{ty}}$I:

$$\dfrac{(\alpha \text{ not in } \Gamma)\ \dfrac{\Gamma \vdash p}{\Gamma \vdash \forall_{\text{ty}} \alpha.\ p}\ \forall_{\text{ty}}\text{I}}{\Gamma[\theta] \vdash (\forall_{\text{ty}} \alpha.\ p)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{(\alpha \text{ not in } \Gamma[\theta])\ \dfrac{\dfrac{\Gamma \vdash p}{\Gamma[\theta] \vdash p[\theta]}\ \text{I.H.}}{\Gamma[\theta] \vdash \forall_{\text{ty}} \alpha.\ p[\theta]}\ \forall\text{I}}{\Gamma[\theta] \vdash (\forall_{\text{ty}} \alpha.\ p)[\theta]}\ \text{defn-subst}$$

Case $\forall_{\text{ty}}$E:

$$\dfrac{\dfrac{\Gamma \vdash [\forall_{\text{ty}}]\ p}{\Gamma \vdash p\ s}\ \forall_{\text{ty}}\text{E}}{\Gamma[\theta] \vdash (p\ s)[\theta]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\Gamma \vdash [\forall_{\text{ty}}]\ p}{\Gamma[\theta] \vdash ([\forall_{\text{ty}}]\ p)[\theta]}\ \text{I.H.}}{\Gamma[\theta] \vdash [\forall_{\text{ty}}]\ p[\theta]}\ \text{defn-subst}}{\Gamma[\theta] \vdash p[\theta]\ s[\theta]}\ \forall_{\text{ty}}\text{E}}{\Gamma[\theta] \vdash (p\ s)[\theta]}\ \text{defn-subst}$$

□

A.5.2 ty-inst

The rule *ty-inst* is *admissible* to $\text{PHOL}_{\xi\eta}$ and PHOL.

$$\dfrac{\Gamma \vdash s}{\Gamma[\phi] \vdash s[\phi]}\ \text{ty-inst}$$

Proof. Induction on the proof-tree of $\Gamma \vdash s$.

Note that in these proofs, if a substitution occurs under a type-lambda (e.g. $(\Lambda\alpha.\ s[\phi])$), we assume that α does not occur in the free-variables of any type resulting from the substitution, as any lambda-term in which this *does* happen can be alpha-renamed such that it doesn't.

Case assm:

$$\dfrac{\dfrac{}{p \vdash p}\ \text{assm}}{p[\phi] \vdash p[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{}{p[\phi] \vdash p[\phi]}\ \text{assm}$$

Case weaken:

$$\dfrac{\dfrac{\Gamma \vdash p}{\Gamma, \Delta \vdash p}\ \text{weaken}}{\Gamma[\phi], \Delta[\phi] \vdash p[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\Gamma \vdash p}{\Gamma[\phi] \vdash p[\phi]}\ \text{I.H.}}{\Gamma[\phi], \Delta[\phi] \vdash p[\phi]}\ \text{weaken}$$

Case contract:

$$\dfrac{\dfrac{\Gamma, \Delta, \Delta \vdash p}{\Gamma, \Delta \vdash p}\ \text{contract}}{\Gamma[\phi], \Delta[\phi] \vdash p[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\Gamma, \Delta, \Delta \vdash p}{\Gamma[\phi], \Delta[\phi], \Delta[\phi] \vdash p[\phi]}\ \text{I.H.}}{\Gamma[\phi], \Delta[\phi] \vdash p[\phi]}\ \text{contract}$$

Case refl:

$$\dfrac{\dfrac{}{\vdash s =_\tau s}\ \text{refl}}{\vdash (s =_\tau s)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{}{\vdash s[\phi] =_{\tau[\phi]} s[\phi]}\ \text{refl}}{\vdash (s =_\tau s)[\phi]}\ \text{refl}$$

Case trans:

$$\dfrac{\dfrac{\Gamma \vdash s =_\tau t \qquad \Delta \vdash t =_\tau u}{\Gamma, \Delta \vdash s =_\tau u}\ \text{trans}}{\Gamma[\phi], \Delta[\phi] \vdash (s =_\tau u)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\Gamma \vdash s =_\tau t}{\Gamma[\phi] \vdash (s =_\tau t)[\phi]}\ \text{I.H.}}{\Gamma[\phi] \vdash s[\phi] =_{\tau[\phi]} t[\phi]}\ \text{defn-subst} \qquad \dfrac{\dfrac{\Delta \vdash t =_\tau u}{\Delta[\phi] \vdash (t =_\tau u)[\phi]}\ \text{I.H.}}{\Delta[\phi] \vdash t[\phi] =_{\tau[\phi]} u[\phi]}\ \text{defn-subst}}{\dfrac{\Gamma[\phi], \Delta[\phi] \vdash s[\phi] =_{\tau[\phi]} u[\phi]}{\Gamma[\phi], \Delta[\phi] \vdash (s =_\tau u)[\phi]}\ \text{defn-subst}}\ \text{trans}$$

Case mk-comb:

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash s = t \qquad \Delta \vdash u = v}{\Gamma, \Delta \vdash s\ u =_{\tau_2} t\ v}\ \text{mk-comb}}{\Gamma[\phi], \Delta[\phi] \vdash (s\ u =_{\tau_2} t\ v)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\Gamma \vdash s = t}{\Gamma[\phi] \vdash (s =_{\tau_1 \Rightarrow \tau_2} t)[\phi]}\ \text{I.H.}}{\Gamma[\phi] \vdash s[\phi] =_{\tau_1[\phi] \Rightarrow \tau_2[\phi]} t[\phi]}\ \text{defn-subst} \qquad \dfrac{\dfrac{\Delta \vdash u = v}{\Gamma[\phi], \Delta[\phi] \vdash (u =_{\tau_1} v)[\phi]}\ \text{I.H.}}{\Gamma[\phi], \Delta[\phi] \vdash u[\phi] =_{\tau_2[\phi]} v[\phi]}\ \text{defn-subst}}{\dfrac{\Gamma[\phi], \Delta[\phi] \vdash s[\phi]\ u[\phi] =_{\tau_2[\phi]} t[\phi]\ v[\phi]}{\Gamma[\phi], \Delta[\phi] \vdash (s\ u =_{\tau_2} t\ v)[\phi]}\ \text{defn-subst}}\ \text{mk-comb}$$

Case beta:

$$\dfrac{\dfrac{}{\vdash (\lambda x.\ s)\ t = s[t/x]}\ \text{beta}}{\vdash \big((\lambda x.\ s)\ t = s[t/x]\big)[\phi]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{}{\vdash (\lambda x_{\sigma[\phi]}.\ s[\phi])\ t[\phi] =_{\tau[\phi]} (s[\phi])[t[\phi]/x_{\sigma[\phi]}]}\ \text{beta}}{\vdash \big((\lambda x.\ s)\ t = s[t/x]\big)[\phi]}\ \text{defn-subst}$$

Case abs:

$$(x_\tau \text{ not in } \Gamma)\ \dfrac{\dfrac{\Gamma \vdash s = t}{\Gamma \vdash (\lambda x_\sigma.\ s) =_\tau (\lambda x_\sigma.\ t)}\ \text{abs}}{\Gamma[\phi_\sigma] \vdash \big((\lambda x_\sigma.\ s) =_\tau (\lambda x_\sigma.\ t)\big)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad (x_{\sigma[\phi]} \text{ not in } \Gamma[\phi])\ \dfrac{\dfrac{\dfrac{\Gamma \vdash s = t}{\Gamma[\phi] \vdash (s =_\tau t)[\phi]}\ \text{I.H.}}{\Gamma[\phi] \vdash s[\phi] =_{\tau[\phi]} t[\phi]}\ \text{defn-subst}}{\Gamma[\phi] \vdash (\lambda x_{\sigma[\phi]}.\ s[\phi]) =_{\tau[\phi]} (\lambda x_{\sigma[\phi]}.\ t[\phi])}\ \text{abs}}{\Gamma[\phi] \vdash \big((\lambda x_\sigma.\ s) =_\tau (\lambda x_\sigma.\ t)\big)[\phi]}\ \text{defn-subst}$$

Case eta:

$$(x_\sigma \text{ not in } t)\ \dfrac{\dfrac{}{\vdash (\lambda x_\sigma.\ t\ x_\sigma) =_{\sigma \Rightarrow \tau} t}\ \text{eta}}{\vdash \big((\lambda x_\sigma.\ t\ x_\sigma) =_{\sigma \Rightarrow \tau} t\big)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad (x_{\sigma[\phi]} \text{ not in } t[\phi])\ \dfrac{\dfrac{}{\vdash (\lambda x_{\sigma[\phi]}.\ t[\phi]\ x_{\sigma[\phi]}) =_{\sigma[\phi] \Rightarrow \tau[\phi]} t[\phi]}\ \text{eta}}{\vdash \big((\lambda x_\sigma.\ t\ x_\sigma) =_{\sigma \Rightarrow \tau} t\big)[\phi]}\ \text{defn-subst}$$

Case tyapp:

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash s =_{\Pi\alpha.\ \tau} t}{\Gamma \vdash s\ [:\sigma:] =_{\tau[\sigma/\alpha]} t\ [:\sigma:]}\ \text{tyapp}}{\Gamma[\phi] \vdash (s\ [:\sigma:] =_{\tau[\sigma/\alpha]} t\ [:\sigma:])[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash s =_{\Pi\alpha.\ \tau} t}{\Gamma[\phi] \vdash (s =_{\Pi\alpha.\ \tau} t)[\phi]}\ \text{I.H.}}{\Gamma[\phi] \vdash s[\phi] =_{(\Pi\alpha.\ \tau[\phi - \alpha])} t[\phi]}\ \text{defn-subst}}{\Gamma[\phi] \vdash s[\phi]\ [:\sigma[\phi]:] =_{\tau[\phi - \alpha, \sigma[\phi]/\alpha]} t[\phi]\ [:\sigma[\phi]:]}\ \text{tyapp}}{\Gamma[\phi] \vdash (s\ [:\sigma:])[\phi] =_{\tau[\sigma/\alpha, \phi]} (t\ [:\sigma:])[\phi]}\ \text{defn-subst}}{\Gamma[\phi] \vdash \big(s\ [:\sigma:] =_{\tau[\sigma/\alpha]} t\ [:\sigma:]\big)[\phi]}\ \text{defn-subst}$$

Case tybeta:

$$\dfrac{\dfrac{}{\vdash (\Lambda\alpha.\ s)\ [:\sigma:] = s[\sigma/\alpha]}\ \text{tybeta}}{\vdash \big((\Lambda\alpha.\ s)\ [:\sigma:] = s[\sigma/\alpha]\big)[\phi]}\ \text{inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{}{\vdash (\Lambda\alpha.\ s[\phi - \alpha])\ [:\sigma[\phi]:] = (s[\phi - \alpha])[\sigma[\phi]/\alpha]}\ \text{tybeta}}{\vdash \big((\Lambda\alpha.\ s)\ [:\sigma:] = s[\sigma/\alpha]\big)[\phi - \alpha][\phi]}\ \text{defn-subst}$$

Case tyabs:

$$\dfrac{(\alpha \text{ not in } \Gamma)\ \dfrac{\Gamma \vdash s =_\tau t}{\vdash (\Lambda\alpha.\ s) =_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)}\ \text{tyabs}}{\Gamma[\phi] \vdash \big((\Lambda\alpha.\ s) =_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)\big)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{(\alpha \text{ not in } \Gamma)\ \dfrac{\dfrac{\Gamma \vdash s =_\tau t}{\Gamma[\phi-\alpha] \vdash (s =_\tau t)[\phi-\alpha]}\ \text{I.H.}}{\Gamma[\phi] \vdash s[\phi-\alpha] =_{\tau[\phi-\alpha]} t[\phi-\alpha]}\ \text{defn-subst}}{\Gamma[\phi] \vdash \big((\Lambda\alpha.\ s) =_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)\big)[\phi]}\ \text{tyabs}\ \text{defn-subst}$$

Case tyeta:

$$\dfrac{(\alpha \text{ not in } t)\ \dfrac{}{\vdash (\Lambda\alpha.\ t\ [\!:\alpha:\!]) =_{\Pi\alpha.\ \tau} t}\ \text{eta}}{\vdash \big((\Lambda\alpha.\ t\ [\!:\alpha:\!]) =_{\Pi\alpha.\ \tau} t\big)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{(\alpha \text{ not in } t[\phi])\ \dfrac{}{\vdash (\Lambda\alpha.\ t[\phi-\alpha]\ [\!:\alpha:\!]) =_{\Pi\alpha.\ \tau[\phi-\alpha]} t[\phi-\alpha]}\ \text{eta}}{(\alpha \text{ not in } t)\ \dfrac{\vdash (\Lambda\alpha.\ t[\phi-\alpha]\ [\!:\alpha[\phi-\alpha]:\!]) =_{\Pi\alpha.\ \tau[\phi-\alpha]} t[\phi]}\ \text{defn-subst}}{\vdash \big((\Lambda\alpha.\ t\ [\!:\alpha:\!]) =_{\Pi\alpha.\ \tau} t\big)[\phi]}\ \text{defn-subst}$$

Case eq-mp:

$$\dfrac{\dfrac{\Gamma \vdash s =_o t \qquad \Delta \vdash s}{\Gamma, \Delta \vdash t}\ \text{eq-mp}}{\Gamma[\phi], \Delta[\phi] \vdash t[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\Gamma \vdash s =_o t}{\Gamma[\phi] \vdash (s =_o t)[\phi]}\ \text{I.H.}}{\Gamma[\phi] \vdash s[\phi] =_o t[\phi]}\ \text{defn-subst} \qquad \dfrac{\Delta \vdash s}{\Delta[\phi] \vdash s[\phi]}\ \text{I.H.}}{\Gamma[\phi], \Delta[\phi] \vdash t[\phi]}\ \text{eq-mp}$$

Case deduct-antisym:

$$\dfrac{\dfrac{\Gamma, q \vdash p \qquad \Delta, p \vdash q}{\Gamma, \Delta \vdash p =_o q}\ \text{deduct-antisym}}{\Gamma[\phi], \Delta[\phi], \vdash (p =_o q)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\Gamma, q \vdash p}{\Gamma[\phi], q[\phi] \vdash p[\phi]}\ \text{I.H.} \qquad \dfrac{\Delta \vdash q}{\Delta[\phi], p[\phi] \vdash q[\phi]}\ \text{I.H.}}{\dfrac{\Gamma[\phi], \Delta[\phi] \vdash p[\phi] =_o q[\phi]}{\Gamma[\phi], \Delta[\phi] \vdash (p =_o q)[\phi]}\ \text{defn-subst}}\ \text{deduct-antisym}$$

Case select-ax:

$$\dfrac{(x \text{ not in } p)\ \dfrac{}{\vdash p\ s \longrightarrow p\ (\varepsilon x_\tau.\ p\ x_\tau)}\ \text{select-ax}}{\vdash (p\ s \longrightarrow p\ (\varepsilon x_\tau.\ p\ x_\tau))[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{(x_{\tau[\phi]} \text{ not in } p[\phi])\ \dfrac{}{\vdash p[\phi]\ s[\phi] \longrightarrow p[\phi]\ (\varepsilon x_{\tau[\phi]}.\ p[\phi]\ x_{\tau[\phi]})}\ \text{select-ax}}{\vdash (p\ x \longrightarrow p\ (\varepsilon x.\ p\ x))[\phi]}\ \text{defn-subst}$$

Case select-ax:

$$\dfrac{\dfrac{}{\vdash \exists(f : \iota \Rightarrow \iota).\ \text{inj}\ f \wedge \neg\, \text{onto}\ f}\ \text{infinity-ax}}{\vdash \big(\exists(f : \iota \Rightarrow \iota).\ \text{inj}\ f \wedge \neg\, \text{onto}\ f\big)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{}{\vdash \exists(f : \iota \Rightarrow \iota).\ \text{inj}\ f \wedge \neg\, \text{onto}\ f}\ \text{infinity-ax}}{\vdash \big(\exists(f : \iota \Rightarrow \iota).\ \text{inj}\ f \wedge \neg\, \text{onto}\ f\big)[\phi]}\ \text{defn-subst}$$

Case $\forall$I:

$$\dfrac{(x \text{ not in } \Gamma)\ \dfrac{\Gamma \vdash p}{\Gamma \vdash \forall x.\ p}\ \forall\text{I}}{\Gamma[\phi] \vdash (\forall x_\tau.\ p)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad (x_{\tau[\phi]} \text{ not in } \Gamma[\phi])\ \dfrac{\dfrac{\Gamma \vdash p}{\Gamma[\phi] \vdash p[\phi]}\ \text{I.H.}}{\dfrac{\Gamma[\phi] \vdash \forall x_{\tau[\phi]}.\ p[\phi]}{\Gamma[\phi] \vdash (\forall x_\tau.\ p)[\phi]}\ \text{defn-subst}}\ \forall\text{I}$$

Case $\forall$E:

$$\dfrac{\dfrac{\Gamma \vdash [\forall]\ p}{\Gamma \vdash p\ s}\ \forall\text{E}}{\Gamma[\phi] \vdash (p\ s)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\Gamma \vdash [\forall]\ p}{\Gamma[\phi] \vdash ([\forall]\ p)[\phi]}\ \text{I.H.}}{\Gamma[\phi] \vdash [\forall]\ p[\phi]}\ \text{defn-subst}}{\dfrac{\Gamma[\phi] \vdash p[\phi]\ s[\phi]}{\Gamma[\phi] \vdash (p\ s)[\phi]}\ \text{defn-subst}}\ \forall\text{E}$$

Case $\forall_{\text{ty}}$I:

$$\dfrac{(\alpha \text{ not in } \Gamma)\ \dfrac{\Gamma \vdash p}{\Gamma \vdash \forall_{\mathsf{ty}}\alpha.\ p}\ \forall_{\mathsf{ty}}\mathrm{I}}{\Gamma[\phi] \vdash (\forall_{\mathsf{ty}}\alpha.\ p)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{(\alpha \text{ not in } \Gamma)\ \dfrac{\dfrac{\Gamma \vdash p}{\Gamma[\phi - \alpha] \vdash p[\phi - \alpha]}\ \text{I.H.}}{\Gamma[\phi] \vdash p[\phi - \alpha]}\ \text{defn-subst}}{(\alpha \text{ not in } \Gamma[\phi - \alpha])\ \dfrac{\Gamma[\phi] \vdash \forall_{\mathsf{ty}}\alpha.\ p[\phi - \alpha]}{\Gamma[\phi] \vdash (\forall_{\mathsf{ty}}\alpha.\ p)[\phi]}\ \text{defn-subst}}\ \forall\mathrm{I}$$

Case $\forall_{\mathsf{ty}}\mathrm{E}$:

$$\dfrac{\dfrac{\Gamma \vdash [\forall_{\mathsf{ty}}]\ p}{\Gamma \vdash p\ s}\ \forall_{\mathsf{ty}}\mathrm{E}}{\Gamma[\phi] \vdash (p\ s)[\phi]}\ \text{ty-inst} \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\Gamma \vdash [\forall_{\mathsf{ty}}]\ p}{\Gamma[\phi] \vdash ([\forall_{\mathsf{ty}}]\ p)[\phi]}\ \text{I.H.}}{\Gamma[\phi] \vdash [\forall_{\mathsf{ty}}]\ p[\phi]}\ \text{defn-subst}}{\Gamma[\phi] \vdash p[\phi]\ s[\phi]}\ \forall_{\mathsf{ty}}\mathrm{E}}{\Gamma[\phi] \vdash (p\ s)[\phi]}\ \text{defn-subst}$$

□

Appendix B

B.1 Translation Definitions

B.1.1 Refl Set

The $\mathcal{R}$ gives the necessary conditions for extensional partial-equivalence to be an equivalence on the translated terms.

$$\mathcal{R}(s) \triangleq \{\!\{x \mathrel{\dot{=}}_\tau x \mid x_\tau \in \mathrm{FV}(s)\}\!\}$$

B.1.2 Eq Set

The Eq set gives the necessary conditions for extensional partial-equivalence on type variables to be translated correctly.

$\mathcal{E}$ on type-variables captures what is needed to ensure that $\dot{=}$ remains a partial-equivalence relation on type variables.

$$\begin{aligned}\mathcal{E}(\alpha) \triangleq\ & \big(\big(\forall x_\alpha\ t_\alpha.\, x \mathrel{\dot{=}}_\alpha y \longrightarrow y \mathrel{\dot{=}}_\alpha x\big)\\ & \wedge \big(\forall x_\alpha\ y_\alpha\ z_\alpha.\, x \mathrel{\dot{=}}_\alpha y \longrightarrow y \mathrel{\dot{=}}_\alpha z \longrightarrow x \mathrel{\dot{=}}_\alpha z\big)\\ & \wedge \big([\dot{=}]_\alpha \mathrel{\dot{=}}_{\alpha\Rightarrow\alpha\Rightarrow o} [\dot{=}]_\alpha\big)\\ & \wedge \exists x.\, x \mathrel{\dot{=}}_\alpha x\big)\end{aligned}$$

This is extended to types in the following manner

$$\mathcal{E}(\tau) \triangleq \{\!\{\mathcal{E}(\alpha) \mid \mathrm{FV}_{\mathrm{ty}}(\tau)\}\!\}$$

and terms similarly

$$\mathcal{E}(s) \triangleq \{\!\{\mathcal{E}(\alpha) \mid \mathrm{FV}_{\mathrm{ty}}(s)\}\!\}$$

B.1.3 Extensional Partial-Equivalence

Extensional partial-equivalence is defined as follows

$$
\begin{array}{lcll}
[\dot{=}]_o & \triangleq & [=]_o & \\
[\dot{=}]_\iota & \triangleq & [=]_\iota & \\
[\dot{=}]_{\tau_1 \Rightarrow \tau_2} & \triangleq & \lambda f\ g.\ \forall x\ y.\, x \mathrel{\dot{=}}_{\tau_1} y \longrightarrow f\ x \mathrel{\dot{=}}_{\tau_2} g\ y & \text{(for } x \text{ and } y \text{ not free in } f \text{ and } g\text{)} \\
[\dot{=}]_{\Pi\alpha.\,\tau} & \triangleq & \lambda f\ g.\ \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow f\ [:\alpha:] \mathrel{\dot{=}}_{\tau} g\ [:\alpha:] & \text{(for } \alpha \text{ not free in } f \text{ and } g\text{)} \\
[\dot{=}]_\alpha & & & \text{(primitive symbol)}
\end{array}
$$

The symbol $[\dot{=}]_\alpha$ has an interaction with type-substitution. When a type is substituted for α, $[\dot{=}]_\alpha$ reduces in the following manner

$$
\begin{gathered}
[\dot{=}]_\alpha[o/\alpha] \triangleq [\dot{=}]_o \\
[\dot{=}]_\alpha[\iota/\alpha] \triangleq [\dot{=}]_\iota \\
[\dot{=}]_\alpha[\tau_1 \Rightarrow \tau_2/\alpha] \triangleq [\dot{=}]_{\tau_1 \Rightarrow \tau_2} \\
[\dot{=}]_\alpha[\Pi\alpha.\,\tau/\alpha] \triangleq [\dot{=}]_{\Pi\alpha.\,\tau} \\
[\dot{=}]_\alpha[\beta/\alpha] \triangleq [\dot{=}]_\beta
\end{gathered}
$$

B.1.4 Translation Definition

Term Translation

The translation is defined for terms as

$$
\begin{array}{lcl}
x^\bullet & \triangleq & x \\
(\lambda x.\ s)^\bullet & \triangleq & \lambda x.\ s^\bullet \\
(s\ t)^\bullet & \triangleq & s^\bullet\ t^\bullet \\
(\Lambda\alpha.\ s)^\bullet & \triangleq & \Lambda\alpha.\ s^\bullet \\
(s\ [:\tau:])^\bullet & \triangleq & s^\bullet\ [:\tau:] \\
([=])^\bullet & \triangleq & [\dot{=}]
\end{array}
$$

Theorem Translation

Given a theorem in $\mathrm{PHOL}_{\xi\eta}$

$$\Gamma \vdash_{\mathrm{PHOL}_{\xi\eta}} p$$

the translation of this theorem into PHOL is

$$\mathcal{E}(\Gamma^\bullet), \mathcal{E}(s^\bullet), \mathcal{R}(\Gamma^\bullet), \mathcal{R}(s^\bullet), \Gamma^\bullet \vdash_{\mathrm{PHOL}} s^\bullet$$

A useful specialisation of this to note is when $s \equiv (t_1 =_\tau t_2)$. Note that $\mathcal{E}([\dot{=}]_\tau) \equiv \mathcal{E}(\tau)$, as the only free $[\dot{=}]_\alpha$ in $[\dot{=}]_\tau$ are the free type variables in τ, and $\mathcal{R}([\dot{=}]_o) \equiv \emptyset$, as there are no free variables in $[\dot{=}]_\tau$.

$$\mathcal{E}(\Gamma^\bullet), \mathcal{E}(\tau), \mathcal{E}(t_1^\bullet), \mathcal{E}(t_2^\bullet), \mathcal{R}(\Gamma^\bullet), \mathcal{R}(t_1^\bullet), \mathcal{R}(t_2^\bullet), \Gamma^\bullet \vdash_{\text{PHOL}} t_1^\bullet \mathrel{\dot{=}}_\tau t_2^\bullet$$

B.2 Translation Lemmas

B.2.1 subst•

Substitutions occasionally occur under the translation operation. In this case, the following rewrite can be applied

$$(s[t/x])^\bullet \equiv s^\bullet[t^\bullet/x]$$

Proof. By induction on s.

Case x:

x is the substituted variable:

$$\begin{aligned}
(x[t/x])^\bullet &\equiv t^\bullet && \text{(defn-subst)} \\
&\equiv x[t^\bullet/x] && \text{(defn•)} \\
&\equiv x^\bullet[t^\bullet/y] && \text{(defn•)}
\end{aligned}$$

x is not the substituted variable:

$$\begin{aligned}
(x[t/y])^\bullet &\equiv x^\bullet && \text{(defn-subst)} \\
&\equiv x && \text{(defn•)} \\
&\equiv x[t^\bullet/y] && \text{(defn-subst)} \\
&\equiv x^\bullet[t^\bullet/y] && \text{(defn•)}
\end{aligned}$$

Case $\lambda x.\ s$:

The substituted x matches the bound x:

$$\begin{aligned}
((\lambda x.\ s)[t/x])^\bullet &\equiv (\lambda x.\ s)^\bullet && \text{(defn-subst)} \\
&\equiv \lambda x.\ s^\bullet && \text{(defn•)} \\
&\equiv (\lambda x.\ s^\bullet)[t^\bullet/x] && \text{(defn-subst)} \\
&\equiv (\lambda x.\ s)^\bullet[t^\bullet/x] && \text{(defn•)}
\end{aligned}$$

The substituted y does not match the bound x:

$$\begin{aligned}
((\lambda x.\ s)[t/y])^\bullet &\equiv (\lambda x.\ s[t/y])^\bullet && \text{(defn-subst)}\\
&\equiv \lambda x.\ (s[t/y])^\bullet && \text{(defn}\bullet\text{)}\\
&\equiv \lambda x.\ (s^\bullet[t^\bullet/y]) && \text{(I.H.)}\\
&\equiv (\lambda x.\ s^\bullet)[t^\bullet/y] && \text{(defn-subst)}\\
&\equiv (\lambda x.\ s)^\bullet[t^\bullet/y] && \text{(defn}\bullet\text{)}
\end{aligned}$$

Case $s\ u$:

$$\begin{aligned}
((s\ u)[t/x])^\bullet &\equiv (s[t/x]\ u[t/x])^\bullet && \text{(defn-subst)}\\
&\equiv (s[t/x])^\bullet\ (u[t/x])^\bullet && \text{(defn}\bullet\text{)}\\
&\equiv (s^\bullet[t^\bullet/x])\ (u^\bullet[t^\bullet/x]) && \text{(I.H.)}\\
&\equiv (s^\bullet\ u^\bullet)[t^\bullet/x] && \text{(defn-subst)}\\
&\equiv (s\ u)^\bullet[t^\bullet/x] && \text{(defn}\bullet\text{)}
\end{aligned}$$

Case $[=]$:

$$\begin{aligned}
([=][t/x])^\bullet &\equiv [=]^\bullet && \text{(defn-subst)}\\
&\equiv [\dot{=}] && \text{(defn}\bullet\text{)}\\
&\equiv [\dot{=}][t^\bullet/x] && \text{(defn-subst)}\\
&\equiv [=]^\bullet[t^\bullet/x] && \text{(defn}\bullet\text{)}
\end{aligned}$$

Case $\Lambda\alpha.\ s$:

$$\begin{aligned}
((\Lambda\alpha.\ s)[t/x])^\bullet &\equiv (\Lambda\alpha.\ s[t/x])^\bullet && \text{(defn-subst)}\\
&\equiv \Lambda\alpha.\ (s[t/x])^\bullet && \text{(defn}\bullet\text{)}\\
&\equiv \Lambda\alpha.\ s^\bullet[t^\bullet/x] && \text{(I.H.)}\\
&\equiv (\Lambda\alpha.\ s^\bullet)[t^\bullet/x] && \text{(defn-subst)}\\
&\equiv (\Lambda\alpha.\ s)^\bullet[t^\bullet/x] && \text{(defn}\bullet\text{)}
\end{aligned}$$

Case $s\ [:\tau:]$:

$$\begin{aligned}
((s\ [:\tau:])[t/x])^\bullet &\equiv ((s[t/x])\ [:\tau:])^\bullet && \text{(defn-subst)}\\
&\equiv (s[t/x])^\bullet\ [:\tau:] && \text{(defn}\bullet\text{)}\\
&\equiv s^\bullet[t^\bullet/x]\ [:\tau:] && \text{(I.H.)}\\
&\equiv (s^\bullet\ [:\tau:])[t^\bullet/x] && \text{(defn-subst)}\\
&\equiv (s\ [:\tau:])^\bullet[t^\bullet/x] && \text{(defn}\bullet\text{)}
\end{aligned}$$

□

B.2.2 tysubst•

Proof. By induction on s.

Case $(x : \tau)$

$$\begin{aligned}
((x:\tau)[\phi])^\bullet &\equiv (x:\tau[\phi])^\bullet && \text{(defn-subst)}\\
&\equiv (x:\tau[\phi]) && \text{(defn}\bullet\text{)}\\
&\equiv (x:\tau)[\phi] && \text{(defn-subst)}\\
&\equiv (x:\tau)^\bullet[\phi] && \text{(defn}\bullet\text{)}
\end{aligned}$$

Case $(\lambda x.\ s)$

$$\begin{aligned}
((\lambda(x:\tau_1).\ s))[\phi])^\bullet &\equiv (\lambda(x:\tau_1[\phi]).\ s[\phi])^\bullet && \text{(defn-subst)}\\
&\equiv \lambda(x:\tau_1[\phi]).\ (s[\phi])^\bullet && \text{(defn}\bullet\text{)}\\
&\equiv \lambda(x:\tau_1[\phi]).\ s^\bullet[\phi] && \text{(I.H.)}\\
&\equiv (\lambda(x:\tau_1).\ s^\bullet)[\phi] && \text{(defn-subst)}\\
&\equiv (\lambda(x:\tau_1).\ s)^\bullet[\phi] && \text{(defn}\bullet\text{)}
\end{aligned}$$

Case $(s\ u)$

$$\begin{aligned}
((s\ u)[\phi])^\bullet &\equiv (s[\phi]\ u[\phi])^\bullet && \text{(defn-subst)}\\
&\equiv (s[\phi])^\bullet\ (u[\phi])^\bullet && \text{(defn}\bullet\text{)}\\
&\equiv (s^\bullet[\phi])\ (u^\bullet[\phi]) && \text{(I.H.)}\\
&\equiv (s^\bullet\ u^\bullet)[\phi] && \text{(defn-subst)}\\
&\equiv (s\ u)^\bullet[\phi] && \text{(defn}\bullet\text{)}
\end{aligned}$$

Case $([=]_\tau)$

$$\begin{aligned}
([=]_\tau[\phi])^\bullet &\equiv ([=]_\tau)^\bullet && \text{(defn-subst)}\\
&\equiv [\dot{=}]_\tau && \text{(defn}\bullet\text{)}\\
&\equiv [\dot{=}]_\tau[\phi] && \text{(defn-subst)}\\
&\equiv [=]^\bullet[\phi] && \text{(defn}\bullet\text{)}
\end{aligned}$$

Case $(\Lambda\alpha.\ s)$

$$\begin{aligned}
((\Lambda\alpha.\ s)[\phi])^\bullet &\equiv (\Lambda\alpha.\ s[\phi - \alpha])^\bullet && \text{(defn-subst)}\\
&\equiv \Lambda\alpha.\ (s[\phi - \alpha])^\bullet && \text{(defn}\bullet\text{)}\\
&\equiv \Lambda\alpha.\ s^\bullet[\phi - \alpha] && \text{(I.H.)}\\
&\equiv (\Lambda\alpha.\ s^\bullet)[\phi] && \text{(defn-subst)}\\
&\equiv (\Lambda\alpha.\ s)^\bullet[\phi] && \text{(defn}\bullet\text{)}
\end{aligned}$$

Case $(s\ [:\tau:])$

$$\begin{aligned}
((s\ [:\tau:])[\phi])^\bullet &\equiv ((s[\phi])\ [:\tau[\phi]:])^\bullet && \text{(defn-subst)}\\
&\equiv (s[\phi])^\bullet\ [:\tau[\phi]:] && \text{(defn}\bullet\text{)}\\
&\equiv s^\bullet[\phi]\ [:\tau[\phi]:] && \text{(I.H.)}\\
&\equiv (s^\bullet\ [:\tau:])[\phi] && \text{(defn-subst)}\\
&\equiv (s\ [:\tau:])^\bullet[\phi] && \text{(defn}\bullet\text{)}
\end{aligned}$$

□

B.2.3 Preservation of Rewrite

$$\frac{\Gamma \vdash s^\bullet =_\tau t^\bullet \qquad \Delta \vdash e^\bullet[s]}{\Gamma, \Delta \vdash e^\bullet[t]}\ \text{rewrite}$$

$$\cfrac{\cfrac{\cfrac{}{\vdash ((\lambda x.\ s)\ t)^\bullet =_\tau ((\lambda x.\ s)\ t)^\bullet}\ \text{refl}}{\vdash ((\lambda x.\ s)\ t)^\bullet =_\tau (\lambda x.\ s^\bullet)\ t^\bullet}\ \text{defn}\bullet}{\vdash ((\lambda x.\ s)\ t)^\bullet =_\tau s^\bullet[t^\bullet/x]}\ \text{rewrite(beta)}$$

B.2.4 Symmetry

Extensional partial-equivalence is symmetric (under sufficient assumptions about partial-equivalence on type variables).

$$\frac{\Gamma \vdash s \mathrel{\dot{=}}_\tau t}{\Gamma, \mathcal{E}(\tau) \vdash t \mathrel{\dot{=}}_\tau s}\ \text{sym}\dot{=}$$

Note that this is stronger than the theorem obtained by translating *sym*, as no $\mathcal{R}$ is needed.

Proof. By induction on τ

(χ: o/ι)

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash s \mathrel{\dot{=}}_{\chi} t}{\Gamma \vdash s =_{\chi} t}\ \text{defn}\dot{=}}{\Gamma \vdash t =_{\chi} s}\ \text{sym}}{\Gamma \vdash t \mathrel{\dot{=}}_{\chi} s}\ \text{defn}\dot{=}}{\Gamma, \mathcal{E}(\tau) \vdash t \mathrel{\dot{=}}_{\chi} s}\ \text{weaken}$$

$(\tau_1 \Rightarrow \tau_2)$

$$[y/x, x/y]\ \dfrac{\dfrac{\dfrac{\Gamma \vdash s \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} t}{\Gamma \vdash \forall x\ y.\, x \mathrel{\dot{=}}_{\tau_1} y \longrightarrow s\ x \mathrel{\dot{=}}_{\tau_2} t\ y}\ \text{defn}\dot{=}}{\Gamma \vdash y \mathrel{\dot{=}}_{\tau_1} x \longrightarrow s\ y \mathrel{\dot{=}}_{\tau_2} t\ x}\ \forall\text{E}^{+} \qquad \dfrac{\dfrac{}{x \mathrel{\dot{=}}_{\tau_1} y \vdash x \mathrel{\dot{=}}_{\tau_1} y}\ \text{assm}}{x \mathrel{\dot{=}}_{\tau_1} y, \mathcal{E}(\tau_1) \vdash y \mathrel{\dot{=}}_{\tau_1} x}\ \text{I.H.}}{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, \mathcal{E}(\tau_1), x \mathrel{\dot{=}}_{\tau_1} y \vdash s\ y \mathrel{\dot{=}}_{\tau_2} t\ x}{\Gamma, \mathcal{E}(\tau_1), \mathcal{E}(\tau_2), x \mathrel{\dot{=}}_{\tau_1} y \vdash t\ x \mathrel{\dot{=}}_{\tau_2} s\ y}\ \text{I.H.}}{\Gamma, \mathcal{E}(\tau_1), \mathcal{E}(\tau_2) \vdash x \mathrel{\dot{=}}_{\tau_1} y \longrightarrow t\ x \mathrel{\dot{=}}_{\tau_2} s\ y}\ \to\text{I}}{\Gamma, \mathcal{E}(\tau_1,), \mathcal{E}(\tau_2) \vdash \forall x\ y.\, x \mathrel{\dot{=}}_{\tau_1} y \longrightarrow t\ x \mathrel{\dot{=}}_{\tau_2} s\ y}\ \forall\text{I}^{+}\ (x, y \text{ fresh})}{\Gamma, \mathcal{E}(\tau_1), \mathcal{E}(\tau_2) \vdash t \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} s}\ \text{defn}\dot{=}}{\Gamma, \mathcal{E}(\tau_1 \Rightarrow \tau_2) \vdash t \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} s}\ \text{defn}\mathcal{E}$$

$(\Lambda\alpha.\ \tau)$

$$[\alpha/\alpha]\ \dfrac{\dfrac{\dfrac{\Gamma \vdash s \mathrel{\dot{=}}_{\Lambda\alpha.\ \tau} t}{\Gamma \vdash \forall\alpha.\, \mathcal{E}(\alpha) \longrightarrow s\ [:\alpha:] \mathrel{\dot{=}}_{\tau} t\ [:\alpha:]}\ \text{defn}\dot{=}}{\Gamma \vdash \mathcal{E}(\alpha) \longrightarrow s\ [:\alpha:] \mathrel{\dot{=}}_{\tau} t\ [:\alpha:]}\ \forall_{\text{ty}}\text{E} \qquad \dfrac{}{\mathcal{E}(\alpha) \vdash \mathcal{E}(\alpha)}\ \text{assm}}{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, \mathcal{E}(\alpha) \vdash s\ [:\alpha:] \mathrel{\dot{=}}_{\tau} t\ [:\alpha:]}{\Gamma, \mathcal{E}(\tau), \mathcal{E}(\alpha) \vdash t\ [:\alpha:] \mathrel{\dot{=}}_{\tau} s\ [:\alpha:]}\ \text{I.H.}}{\Gamma, \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\alpha) \vdash t\ [:\alpha:] \mathrel{\dot{=}}_{\tau} s\ [:\alpha:]}\ \text{defn}\mathcal{E} + \text{contract}^{*}}{\Gamma, \mathcal{E}(\Pi\alpha.\ \tau) \vdash \mathcal{E}(\alpha) \longrightarrow t\ [:\alpha:] \mathrel{\dot{=}}_{\tau} s\ [:\alpha:]}\ \to\text{I}}{\Gamma, \mathcal{E}(\Pi\alpha.\ \tau) \vdash \forall\alpha.\, \mathcal{E}(\alpha) \longrightarrow t\ [:\alpha:] \mathrel{\dot{=}}_{\tau} s\ [:\alpha:]}\ \forall_{\text{ty}}\text{I}\ (\alpha \text{ fresh})}{\Gamma, \mathcal{E}(\Pi\alpha.\ \tau) \vdash t \mathrel{\dot{=}}_{\Pi\alpha.\ \tau} s}\ \text{defn}\dot{=}$$

(α)

$$[t/x, s/y]\ \dfrac{\dfrac{\dfrac{}{\mathcal{E}(\alpha) \vdash \forall x_\alpha\ y_\alpha.\ x \stackrel{\bullet}{=}_\alpha y \longrightarrow y \stackrel{\bullet}{=}_\alpha x}\ \text{defn}\mathcal{E} + \wedge\text{E}^+}{\mathcal{E}(\alpha) \vdash t \stackrel{\bullet}{=}_\alpha s \longrightarrow s \stackrel{\bullet}{=}_\alpha t}\ \forall\text{E + beta} \qquad \Gamma \vdash t \stackrel{\bullet}{=}_\alpha s}{\Gamma, \mathcal{E}(\alpha) \vdash s \stackrel{\bullet}{=}_\alpha t}\ \rightarrow\text{E}$$

□

B.2.5 Reflexive Terms

It is necessary for the proof to have a term at each (non-variable) type, which is reflexive with respect to the $\stackrel{\bullet}{=}$ relation. (This idea comes from Gandy's translation [Gandy, 1956].)

We define Z_τ as

$$\begin{aligned}
\text{Z}_o &\triangleq \top \\
\text{Z}_\iota &\triangleq z && \text{(for some } z \text{ in } \iota\text{)} \\
\text{Z}_{\tau \Rightarrow \sigma} &\triangleq \lambda x.\ \text{Z}_\sigma && \text{(where } x \notin \text{FV}(Z_\sigma)\text{)} \\
\text{Z}_{\Pi\alpha.\ \tau} &\triangleq \Lambda\alpha.\ \text{Z}_\tau \\
\text{Z}_\alpha &\triangleq \varepsilon z.\ z \stackrel{\bullet}{=}_\alpha z
\end{aligned}$$

Note that, as is traditional, we assume that the domain of every type is non-empty.

The following theorem holds

$$\dfrac{}{\mathcal{E}(\text{Z}_\tau) \vdash \text{Z}_\tau \stackrel{\bullet}{=}_\tau \text{Z}_\tau}\ \text{reflZ}$$

Proof. By induction on τ

Case o:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\top \vdash \top}\ \text{assm} \quad \dfrac{}{\top \vdash \top}\ \text{assm}}{\vdash \top = \top}\ \text{deduct-antisym}}{\vdash \top \stackrel{\bullet}{=} \top}\ \text{defn}\stackrel{\bullet}{=}}{\mathcal{E}(\top) \vdash \top \stackrel{\bullet}{=} \top}\ \text{defn}\mathcal{E}}{\mathcal{E}(Z_o) \vdash \text{Z}_o \stackrel{\bullet}{=}_o \text{Z}_o}\ \text{defnZ}$$

Case ι:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash z =_\iota z}\,\text{refl}}{\vdash z \mathrel{\dot{=}}_\iota z}\,\text{defn}\dot{=}}{\mathcal{E}(z) \vdash z \mathrel{\dot{=}}_\iota z}\,\text{defn}\mathcal{E}}{\mathcal{E}(Z_\iota) \vdash \mathrm{Z}_\iota \mathrel{\dot{=}}_\iota \mathrm{Z}_\iota}\,\text{defnZ}$$

Case $\tau \Rightarrow \sigma$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{E}(Z_\sigma) \vdash Z_\sigma \mathrel{\dot{=}}_\sigma \mathrm{Z}_\sigma}\,\text{I.H.}}{\mathcal{E}(Z_\sigma) \vdash Z_\sigma[x/x] \mathrel{\dot{=}}_\sigma \mathrm{Z}_\sigma[y/x]}\,\text{defn-subst}\ (x \notin \mathrm{FV}(\mathrm{Z}_\sigma))}{\mathcal{E}(Z_\sigma) \vdash (\lambda x.\ Z_\sigma)\ x \mathrel{\dot{=}}_\sigma (\lambda x.\ \mathrm{Z}_\sigma)\ y}\,\text{beta}}{\mathcal{E}(Z_\sigma), x \mathrel{\dot{=}}_\tau y \vdash (\lambda x.\ Z_\sigma)\ x \mathrel{\dot{=}}_\sigma (\lambda x.\ \mathrm{Z}_\sigma)\ y}\,\text{weaken}}{\mathcal{E}(Z_\sigma) \vdash \forall x\ y.\ x \mathrel{\dot{=}}_\tau y \longrightarrow (\lambda x.\ Z_\sigma)\ x \mathrel{\dot{=}}_\sigma (\lambda x.\ \mathrm{Z}_\sigma)\ y}\,\forall\mathrm{E}^+ + \to\mathrm{E}}{\mathcal{E}(Z_\sigma) \vdash (\lambda x.\ Z_\sigma) \mathrel{\dot{=}}_{\tau\Rightarrow\sigma} (\lambda x.\ \mathrm{Z}_\sigma)}\,\text{defn}\dot{=}}{\mathcal{E}(\lambda x.\ Z_\sigma) \vdash (\lambda x.\ Z_\sigma) \mathrel{\dot{=}}_{\tau\Rightarrow\sigma} (\lambda x.\ \mathrm{Z}_\sigma)}\,\text{defn}\mathcal{E}}{\mathcal{E}(\mathrm{Z}_{\tau\Rightarrow\sigma}) \vdash \mathrm{Z}_{\tau\Rightarrow\sigma} \mathrel{\dot{=}}_{\tau\Rightarrow\sigma} \mathrm{Z}_{\tau\Rightarrow\sigma}}\,\text{defnZ}$$

Case $\Pi\alpha.\ \tau$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{E}(\mathrm{Z}_\tau) \vdash \mathrm{Z}_\tau \mathrel{\dot{=}}_\tau \mathrm{Z}_\tau}\,\text{I.H.}}{\mathcal{E}(\mathrm{Z}_\tau) \vdash \mathrm{Z}_\tau[\alpha/\alpha] \mathrel{\dot{=}}_\tau \mathrm{Z}_\tau[\alpha/\alpha]}\,\text{defn-subst}}{\mathcal{E}(\mathrm{Z}_\tau) \vdash (\Lambda\alpha.\ \mathrm{Z}_\tau)\ [:\alpha:] \mathrel{\dot{=}}_\tau (\Lambda\alpha.\ \mathrm{Z}_\tau)\ [:\alpha:]}\,\text{tybeta}}{\mathcal{E}(\Lambda\alpha.\ \mathrm{Z}_\tau), \mathcal{E}(\alpha) \vdash (\Lambda\alpha.\ \mathrm{Z}_\tau)\ [:\alpha:] \mathrel{\dot{=}}_\tau (\Lambda\alpha.\ \mathrm{Z}_\tau)\ [:\alpha:]}\,\text{defn}\mathcal{E} + \text{weaken}^*}{\mathcal{E}(\Lambda\alpha.\ \mathrm{Z}_\tau) \vdash \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ \mathrm{Z}_\tau)\ [:\alpha:] \mathrel{\dot{=}}_\tau (\Lambda\alpha.\ \mathrm{Z}_\tau)\ [:\alpha:]}\,\to\mathrm{I}}{\mathcal{E}(\Lambda\alpha.\ \mathrm{Z}_\tau) \vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ \mathrm{Z}_\tau)\ [:\alpha:] \mathrel{\dot{=}}_\tau (\Lambda\alpha.\ \mathrm{Z}_\tau)\ [:\alpha:]}\,\forall_{\mathsf{ty}}\mathrm{I}\ (\alpha \text{ fresh})}{\mathcal{E}(\Lambda\alpha.\ \mathrm{Z}_\tau) \vdash (\Lambda\alpha.\ \mathrm{Z}_\tau) \mathrel{\dot{=}}_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ \mathrm{Z}_\tau)}\,\text{defn}\dot{=}}{\mathcal{E}(\mathrm{Z}_{\Pi\alpha.\ \tau}) \vdash \mathrm{Z}_{\Pi\alpha.\ \tau} \mathrel{\dot{=}}_{\Pi\alpha.\ \tau} \mathrm{Z}_{\Pi\alpha.\ \tau}}\,\text{defnZ}$$

Case α:

$$\dfrac{\dfrac{}{\vdash z \mathrel{\dot{=}}_\alpha z \longrightarrow (\varepsilon z.\ z \mathrel{\dot{=}}_\alpha z) \mathrel{\dot{=}}_\alpha (\varepsilon z.\ z \mathrel{\dot{=}}_\alpha z)}\,\text{select-ax} \qquad (\text{fresh } z)\ \dfrac{\dfrac{}{\mathcal{E}(\mathrm{Z}_\alpha) \vdash \exists x.\ x \mathrel{\dot{=}}_\alpha x}\,\text{defn}\mathcal{E} + \wedge\mathrm{E}^+}{\mathcal{E}(\mathrm{Z}_\alpha) \vdash z \mathrel{\dot{=}}_\alpha z}\,\exists\mathrm{E}}{[\mathrm{Z}_\alpha/x]\ \dfrac{\mathcal{E}(\mathrm{Z}_\alpha) \vdash (\varepsilon z.\ z \mathrel{\dot{=}}_\alpha z) \mathrel{\dot{=}}_\alpha (\varepsilon z.\ z \mathrel{\dot{=}}_\alpha z)}{\mathcal{E}(\mathrm{Z}_\alpha) \vdash \mathrm{Z}_\alpha \mathrel{\dot{=}}_\alpha \mathrm{Z}_\alpha}\,\text{defnZ}}\,\to\mathrm{I}$$

□

B.2.6 drop-unused$\mathcal{R}$

$$(\overline{x} \cap (\mathrm{FV}(\Gamma) \cup \mathrm{FV}(p)) = \emptyset)\ \frac{\mathcal{R}(\overline{x}), \Gamma \vdash p}{\mathcal{E}(\overline{x}), \Gamma \vdash p}\ \text{drop-unused}\mathcal{R}$$

Proof. Firstly, define θ_{Z} to be $\mathrm{Z}_{\tau_1}/x_{1\tau_1}, \mathrm{Z}_{\tau_2}/x_{2\tau_2}, \ldots, \mathrm{Z}_{\tau_n}/x_{n\tau_n}$ for each $x_i \in \overline{x}$.

$$\dfrac{\dfrac{}{\mathcal{E}(\overline{x}) \vdash^* \mathcal{R}(\overline{x})[\theta_{\mathrm{Z}}]}\ \text{refl}\mathrm{Z}^* \qquad (\overline{x} \cap (\mathrm{FV}(\Gamma) \cup \mathrm{FV}(p)) = \emptyset)\ \dfrac{\dfrac{\mathcal{R}(\overline{x}), \Gamma \vdash p}{\mathcal{R}(\overline{x})[\theta_{\mathrm{Z}}], \Gamma[\theta_{\mathrm{Z}}] \vdash p[\theta_{\mathrm{Z}}]}\ \text{inst}}{\mathcal{R}(\overline{x})[\theta_{\mathrm{Z}}], \Gamma \vdash p}\ \text{defn-subst}}{\mathcal{E}(\overline{x}), \Gamma \vdash p}\ \text{ante-subst}^*$$

□

B.2.7 Eq-Set on Propositions

$$\frac{}{\vdash \mathcal{E}(\alpha)[o/\alpha]}\ \mathcal{E}\text{-on-}o$$

Proof.

$$\mathcal{E}(\alpha)[o/\alpha]$$

By defn$\mathcal{E}$

$$\begin{aligned}
&\equiv \Big(\big(\forall(s,t,u:\alpha).\ s \doteq_\alpha t \longrightarrow t \doteq_\alpha u \longrightarrow s \doteq_\alpha u \big) \\
&\quad \wedge \big(\forall(s:\alpha).\ s \doteq_\alpha s \big) \\
&\quad \wedge \big(\forall(s,t:\alpha).\ s \doteq_\alpha t \longrightarrow t \doteq_\alpha s \big) \\
&\quad \wedge \big([\doteq]_\alpha \doteq_{\alpha\Rightarrow\alpha\Rightarrow o} [\doteq]_\alpha \big) \Big)[o/\alpha]
\end{aligned}$$

By defn-subst

$$\begin{aligned}
&\equiv \big(\forall(s,t,u:o).\ s \doteq_o t \longrightarrow t \doteq_o u \longrightarrow s \doteq_o u \big) \\
&\quad \wedge \big(\forall(s:o).\ s \doteq_o s \big) \\
&\quad \wedge \big(\forall(s,t:o).\ s \doteq_o t \longrightarrow t \doteq_o s \big) \\
&\quad \wedge \big([\doteq]_o \doteq_{o\Rightarrow o\Rightarrow o} [\doteq]_o \big)
\end{aligned}$$

The first three of the conjuncts are simple to derive from *trans*, *refl*, and *sym*. We can prove the last as follows

$$\dfrac{\dfrac{\dfrac{}{x_1 \mathrel{\dot{=}}_o x_2 \vdash x_1 \mathrel{\dot{=}}_o x_2}\ \text{assm}}{x_1 \mathrel{\dot{=}}_o x_2 \vdash x_1 =_o x_2}\ \text{defn}\dot{=} \quad \dfrac{\dfrac{}{y_1 \mathrel{\dot{=}}_o y_2 \vdash y_1 \mathrel{\dot{=}}_o y_2}\ \text{assm}}{y_1 \mathrel{\dot{=}}_o y_2 \vdash y_1 =_o y_2}\ \text{defn}\dot{=} \quad \dfrac{}{\vdash ([=]_o\ x_1\ y_1) =_o ([=]_o\ x_1\ y_1)}\ \text{refl}}{\dfrac{\dfrac{\dfrac{\dfrac{x_1 \mathrel{\dot{=}}_o x_2, y_1 \mathrel{\dot{=}}_o y_2 \vdash ([=]_o\ x_1\ y_1) =_o ([=]_o\ x_2\ y_2)}{x_1 \mathrel{\dot{=}}_o x_2, y_1 \mathrel{\dot{=}}_o y_2 \vdash ([\dot{=}]_o\ x_1\ y_1) \mathrel{\dot{=}}_o ([\dot{=}]_o\ x_2\ y_2)}\ \text{defn}\dot{=}}{x_1 \mathrel{\dot{=}}_o x_2 \vdash y_1 \mathrel{\dot{=}}_o y_2 \longrightarrow ([\dot{=}]_o\ x_1\ y_1) \mathrel{\dot{=}}_o ([\dot{=}]_o\ x_2\ y_2)}\ \to\text{I}}{(x_1, x_2 \text{ fresh})\quad x_1 \mathrel{\dot{=}}_o x_2 \vdash \forall y_1\ y_2.\, y_1 \mathrel{\dot{=}}_o y_2 \longrightarrow ([\dot{=}]_o\ x_1\ y_1) \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} ([\dot{=}]_o\ x_2\ y_2)}\ \forall\text{I}^+}{\dfrac{\dfrac{x_1 \mathrel{\dot{=}}_o x_2 \vdash ([\dot{=}]_o\ x_1) \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} ([\dot{=}]_o\ x_2)}{\vdash x_1 \mathrel{\dot{=}}_o x_2 \longrightarrow ([\dot{=}]_o\ x_1) \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} ([\dot{=}]_o\ x_2)}\ \to\text{I}}{(x_1, x_2 \text{ fresh})\quad \vdash \forall x_1\ x_2.\, x_1 \mathrel{\dot{=}}_o x_2 \longrightarrow ([\dot{=}]_o\ x_1) \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} ([\dot{=}]_o\ x_2)}\ \forall\text{I}^+}{\vdash [\dot{=}]_o \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} [\dot{=}]_o}\ \text{defn}\dot{=}}\ \text{defn}\dot{=}}\ \text{rewrite}^+$$

□

B.2.8 drop-unused$\mathcal{E}$

$$(\overline{\alpha} \cap (\mathrm{FV}_{\mathrm{ty}}(\Gamma) \cup \mathrm{FV}_{\mathrm{ty}}(p)) = \emptyset)\ \frac{\mathcal{E}(\overline{\alpha}), \Gamma \vdash p}{\Gamma \vdash p}\ \text{drop-unused}\mathcal{E}$$

Proof. Firstly, define ϕ to be $o/\alpha_1, o/\alpha_2, \ldots, o/\alpha_n$ for each $\alpha_i \in \overline{\alpha}$.

$$\dfrac{\dfrac{}{\vdash^* \mathcal{E}(\overline{\alpha})[\phi_o]}\ \mathcal{E}\text{-on-}o^* \quad (\overline{\alpha} \cap (\mathrm{FV}_{\mathrm{ty}}(\Gamma) \cup \mathrm{FV}_{\mathrm{ty}}(p)) = \emptyset)\ \dfrac{\dfrac{\mathcal{E}(\overline{\alpha}), \Gamma \vdash p}{\mathcal{E}(\overline{\alpha})[\phi_o], \Gamma[\phi_o] \vdash p[\phi_o]}\ \text{tyinst}}{\mathcal{E}(\overline{\alpha})[\phi_o], \Gamma \vdash p}\ \text{defn-subst}}{\Gamma \vdash p}\ \text{ante-subst}^*$$

□

B.2.9 Translation of the Connectives

$$
\begin{aligned}
\top &\triangleq (\lambda p.\ p) = (\lambda p.\ p) \\
[\wedge] &\triangleq \lambda p\ q.\ (\lambda f.\ f\ p\ q) = (\lambda f.\ f\ \top\ \top) \\
p \wedge q &\triangleq [\wedge]\ p\ q \\
[\longrightarrow] &\triangleq \lambda p\ q.\ (p \wedge q) = p \\
p \longrightarrow q &\triangleq [\longrightarrow]\ p\ q \\
[\forall]_\tau &\triangleq \lambda p.\ p =_\tau (\lambda x.\ \top) \\
\forall x.\, p &\triangleq [\forall]\ (\lambda x.\ p) && \text{(where } p \text{ may contain } x\text{)} \\
[\exists] &\triangleq \lambda p.\ \forall z_o.(\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o \\
\exists x.\, p &\triangleq [\exists]\ (\lambda x.\ p) && \text{(where } p \text{ may contain } x\text{)} \\
[\vee] &\triangleq \lambda p\ q.\ \forall z_o.(p \longrightarrow z_o) \longrightarrow (q \longrightarrow z_o) \longrightarrow z_o \\
p \vee q &\triangleq [\vee]\ p\ q \\
\bot &\triangleq \forall z_o.\, z_o \\
[\neg] &\triangleq \lambda p.\ p \longrightarrow \bot \\
\neg p &\triangleq [\neg]\ p \\
[\forall_{\mathsf{ty}}] &\triangleq \lambda z.\ z =_{\Pi\alpha.\ o} (\Lambda\alpha.\ \top) \\
\forall_{\mathsf{ty}}\alpha.\ p &\triangleq [\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p) && \text{(where } p \text{ may contain } \alpha\text{)}
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{inj} &\triangleq \lambda f.\ \forall x\ y.\, f\ x = f\ y \longrightarrow x = y \\
\mathrm{onto} &\triangleq \lambda f.\ \forall y.\, \exists x.\, f\ x = y
\end{aligned}
$$

Truth

$$
\dfrac{\dfrac{\dfrac{}{\vdash (\lambda p.\ p) =_{o\Rightarrow o} (\lambda p.\ p)}\ \text{refl}}{(\lambda p.\ p) \doteq_{o\Rightarrow o} (\lambda p.\ p) \vdash (\lambda p.\ p) =_{o\Rightarrow o} (\lambda p.\ p)}\ \text{weaken} \qquad (\text{fresh } x, y)\ \dfrac{\dfrac{\dfrac{\dfrac{}{x =_o y \vdash x =_o y}\ \text{assm}}{x =_o y \vdash (\lambda p.\ p)\ x =_o (\lambda p.\ p)\ y}\ \to\text{I}}{\vdash \forall x\ y.\, x =_o y \longrightarrow (\lambda p.\ p)\ x =_o (\lambda p.\ p)\ y}\ \forall\text{I}^+}{\vdash (\lambda p.\ p) \doteq_{o\Rightarrow o} (\lambda p.\ p)}\ \text{defn}\doteq}{(\lambda p.\ p) =_{o\Rightarrow o} (\lambda p.\ p) \vdash (\lambda p.\ p) \doteq_{o\Rightarrow o} (\lambda p.\ p)}\ \text{weaken}}{\dfrac{\vdash ((\lambda p.\ p) \doteq_{o\Rightarrow o} (\lambda p.\ p)) =_o ((\lambda p.\ p) =_{o\Rightarrow o} (\lambda p.\ p))}{\vdash ((\lambda p.\ p) =_{o\Rightarrow o} (\lambda p.\ p))^\bullet =_o ((\lambda p.\ p) =_{o\Rightarrow o} (\lambda p.\ p))}\ \text{defn}\bullet}{\vdash \top^\bullet =_o \top}\ \text{defn}\top}\ \text{deduct-antisym}
$$

Conjunction

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x_1 \mathrel{\dot{=}}_o x_2 \vdash x_1 \mathrel{\dot{=}}_o x_2}\,\text{assm}}{x_1 \mathrel{\dot{=}}_o x_2, y_1 \mathrel{\dot{=}}_o y_2 \vdash x_1 \mathrel{\dot{=}}_o x_2}\,\text{weaken}}{x_1 \mathrel{\dot{=}}_o x_2, y_1 \mathrel{\dot{=}}_o y_2 \vdash (\lambda y.\ x_1)\ y_1 \mathrel{\dot{=}}_o (\lambda y.\ x_2)\ y_2}\,\text{beta}}{x_1 \mathrel{\dot{=}}_o x_2 \vdash y_1 \mathrel{\dot{=}}_o y_2 \longrightarrow (\lambda y.\ x_1)\ y_1 \mathrel{\dot{=}}_o (\lambda y.\ x_2)\ y_2}\,{\to}\text{I}}{(\text{fresh } y_1, y_2)\ \dfrac{x_1 \mathrel{\dot{=}}_o x_2 \vdash \forall y_1\ y_2.\ y_1 \mathrel{\dot{=}}_o y_2 \longrightarrow (\lambda y.\ x_1)\ y_1 \mathrel{\dot{=}}_o (\lambda y.\ x_2)\ y_2}{x_1 \mathrel{\dot{=}}_o x_2 \vdash (\lambda y.\ x_1) \mathrel{\dot{=}}_{o\Rightarrow o} (\lambda y.\ x_2)}\,\text{beta}}{x_1 \mathrel{\dot{=}}_o x_2 \vdash (\lambda x\ y.\ x)\ x_1 \mathrel{\dot{=}}_{o\Rightarrow o} (\lambda x\ y.\ x)\ x_2}\,{\to}\text{I}}{(\text{fresh } x_1, x_2)\ \dfrac{\vdash x_1 \mathrel{\dot{=}}_o x_2 \longrightarrow (\lambda x\ y.\ x)\ x_1 \mathrel{\dot{=}}_{o\Rightarrow o} (\lambda x\ y.\ x)\ x_2}{\vdash \forall x_1\ x_2.\ x_1 \mathrel{\dot{=}}_o x_2 \longrightarrow (\lambda x\ y.\ x)\ x_1 \mathrel{\dot{=}}_{o\Rightarrow o} (\lambda x\ y.\ x)\ x_2}\,\text{defn}\dot{=}}{\vdash (\lambda x\ y.\ x) \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} (\lambda x\ y.\ x)}\,\text{defn}\dot{=}$$

$$\vdots\ 1.1.1$$

$$[\theta]\ \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(p \wedge q)^\bullet \vdash (p \wedge q)^\bullet}\,\text{assm}}{(p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet) \mathrel{\dot{=}}_{(o\Rightarrow o\Rightarrow o)\Rightarrow o} (\lambda f.\ f\ \top\ \top)}\,\text{defn}\bullet + \text{defn}\wedge}{(p \wedge q)^\bullet \vdash \forall f\ g.\ f \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} g \longrightarrow (\lambda f.\ f\ p^\bullet\ q^\bullet)\ f \mathrel{\dot{=}}_o (\lambda f.\ f\ \top\ \top)\ g}\,\text{defn}\dot{=}}{(p \wedge q)^\bullet \vdash (\lambda x\ y.\ x) \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} (\lambda x\ y.\ x) \longrightarrow (\lambda f.\ f\ p^\bullet\ q^\bullet)\ (\lambda x\ y.\ x) \mathrel{\dot{=}}_o (\lambda f.\ f\ \top\ \top)\ (\lambda x\ y.\ x)}\,\forall\text{E}^+ \qquad \begin{array}{c}\vdots\ 1.1.1\\ \vdash (\lambda x\ y.\ x) \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} (\lambda x\ y.\ x)\end{array}}{(p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet)\ (\lambda x\ y.\ x) \mathrel{\dot{=}}_o (\lambda f.\ f\ \top\ \top)\ (\lambda x\ y.\ x)}\,{\to}\text{E}}{(p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet)\ (\lambda x\ y.\ x) =_o (\lambda f.\ f\ \top\ \top)\ (\lambda x\ y.\ x)}\,\text{defn}\dot{=}}{(p \wedge q)^\bullet \vdash ((\lambda x\ y.\ x)\ p^\bullet\ q^\bullet) =_o ((\lambda x\ y.\ x)\ \top\ \top)}\,\text{beta}}{(p \wedge q)^\bullet \vdash p^\bullet =_o \top}\,\text{beta}$$

$$\vdots\ 1.1$$

Where $\theta \triangleq (\lambda x\ y.\ x)/f, (\lambda x\ y.\ x)/g$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{y_1 \mathrel{\dot{=}}_o y_2 \vdash y_1 \mathrel{\dot{=}}_o y_2}\,\text{assm}}{x_1 \mathrel{\dot{=}}_o x_2, y_1 \mathrel{\dot{=}}_o y_2 \vdash y_1 \mathrel{\dot{=}}_o y_2}\,\text{weaken}}{x_1 \mathrel{\dot{=}}_o x_2, y_1 \mathrel{\dot{=}}_o y_2 \vdash (\lambda y.\ y)\ y_1 \mathrel{\dot{=}}_o (\lambda y.\ y)\ y_2}\,\text{beta}}{x_1 \mathrel{\dot{=}}_o x_2 \vdash y_1 \mathrel{\dot{=}}_o y_2 \longrightarrow (\lambda y.\ y)\ y_1 \mathrel{\dot{=}}_o (\lambda y.\ y)\ y_2}\,{\to}\text{I}}{(\text{fresh } y_1, y_2)\ \dfrac{x_1 \mathrel{\dot{=}}_o x_2 \vdash \forall y_1\ y_2.\ y_1 \mathrel{\dot{=}}_o y_2 \longrightarrow (\lambda y.\ y)\ y_1 \mathrel{\dot{=}}_o (\lambda y.\ y)\ y_2}{x_1 \mathrel{\dot{=}}_o x_2 \vdash (\lambda y.\ y) \mathrel{\dot{=}}_{o\Rightarrow o} (\lambda y.\ y)}\,\text{beta}}{x_1 \mathrel{\dot{=}}_o x_2 \vdash (\lambda x\ y.\ y)\ x_1 \mathrel{\dot{=}}_{o\Rightarrow o} (\lambda x\ y.\ y)\ x_2}\,{\to}\text{I}}{(\text{fresh } x_1, x_2)\ \dfrac{\vdash x_1 \mathrel{\dot{=}}_o x_2 \longrightarrow (\lambda x\ y.\ y)\ x_1 \mathrel{\dot{=}}_{o\Rightarrow o} (\lambda x\ y.\ y)\ x_2}{\vdash \forall x_1\ x_2.\ x_1 \mathrel{\dot{=}}_o x_2 \longrightarrow (\lambda x\ y.\ y)\ x_1 \mathrel{\dot{=}}_{o\Rightarrow o} (\lambda x\ y.\ y)\ x_2}\,\text{defn}\dot{=}}{\vdash (\lambda x\ y.\ y) \mathrel{\dot{=}}_{o\Rightarrow o\Rightarrow o} (\lambda x\ y.\ y)}\,\text{defn}\dot{=}$$

$$\vdots\ 1.2.1$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(p \wedge q)^\bullet \vdash (p \wedge q)^\bullet}\,\text{assm}}{(p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet) \doteq_{(o \Rightarrow o \Rightarrow o) \Rightarrow o} (\lambda f.\ f\ \top\ \top)}\,\text{defn}^\bullet + \text{defn}^\wedge}{(p \wedge q)^\bullet \vdash \forall f\ g.\ f \doteq_{o \Rightarrow o \Rightarrow o} g \longrightarrow (\lambda f.\ f\ p^\bullet\ q^\bullet)\ f \doteq_o (\lambda f.\ f\ \top\ \top)\ g}\,\text{defn}^\doteq}{(p \wedge q)^\bullet \vdash f \doteq_{o \Rightarrow o \Rightarrow o} g \longrightarrow (\lambda f.\ f\ p^\bullet\ q^\bullet)\ (\lambda x\ y.\ y) \doteq_o (\lambda f.\ f\ \top\ \top)\ (\lambda x\ y.\ y)}\,\forall\text{E}^+\ [\theta] \qquad \vdots\,1.2.1 \quad \vdash (\lambda x\ y.\ y) \doteq_{o \Rightarrow o \Rightarrow o} (\lambda x\ y.\ y)}{(p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet)\ (\lambda x\ y.\ y) \doteq_o (\lambda f.\ f\ \top\ \top)\ (\lambda x\ y.\ y)}\,\rightarrow\text{I}$$

$$\dfrac{\dfrac{\dfrac{(p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet)\ (\lambda x\ y.\ y) =_o (\lambda f.\ f\ \top\ \top)\ (\lambda x\ y.\ y)}{(p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet)\ (\lambda x\ y.\ y) \doteq_o (\lambda f.\ f\ \top\ \top)\ (\lambda x\ y.\ y)}\,\text{defn}^\doteq}{(p \wedge q)^\bullet \vdash ((\lambda x\ y.\ y)\ p^\bullet\ q^\bullet) =_o ((\lambda x\ y.\ y)\ \top\ \top)}\,\text{beta}}{(p \wedge q)^\bullet \vdash p^\bullet =_o \top}\,\text{beta}$$

$$\vdots\,1.2$$

Where $\theta \triangleq (\lambda x\ y.\ y)/f, (\lambda x\ y.\ y)/g$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdots\,1.1}{(p \wedge q)^\bullet \vdash p^\bullet = \top}}{(p \wedge q)^\bullet \vdash \top = p^\bullet}\,\text{sym} \quad \dfrac{\vdots\,1.2}{(p \wedge q)^\bullet \vdash \top = q^\bullet}\,\text{sym} \quad \dfrac{}{\vdash (\lambda f.\ f\ \top\ \top) =_{(o \Rightarrow o \Rightarrow o) \Rightarrow o} (\lambda f.\ f\ \top\ \top)}\,\text{refl}}{(p \wedge q)^\bullet, (p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet) =_{(o \Rightarrow o \Rightarrow o) \Rightarrow o} (\lambda f.\ f\ \top\ \top)}\,\text{rewrite}^+}{(p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet) =_{(o \Rightarrow o \Rightarrow o) \Rightarrow o} (\lambda f.\ f\ \top\ \top)}\,\text{contract}$$

$$\vdots\,1$$

(n.b. f not in $p^\bullet, q^\bullet$)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{f \doteq_{o \Rightarrow o \Rightarrow o} g \vdash f \doteq_{o \Rightarrow o \Rightarrow o} g}\,\text{assm}}{f \doteq_{o \Rightarrow o \Rightarrow o} g \vdash \forall x\ y x =_o y \longrightarrow f\ x \doteq_{o \Rightarrow o} g\ y}\,\text{defn}^\doteq}{f \doteq_{o \Rightarrow o \Rightarrow o} g \vdash \top =_o \top \longrightarrow f\ \top \doteq_{o \Rightarrow o} g\ \top}\,\forall\text{E}^+ \quad \dfrac{}{\vdash \top =_o \top}\,\text{refl}}{f \doteq_{o \Rightarrow o \Rightarrow o} g \vdash f\ \top \doteq_{o \Rightarrow o} g\ \top}\,\rightarrow\text{E}}{f \doteq_{o \Rightarrow o \Rightarrow o} g \vdash \forall x\ y.\ x =_o y \longrightarrow f\ \top\ x =_o g\ \top\ y}\,\text{defn}^\doteq}{f \doteq_{o \Rightarrow o \Rightarrow o} g \vdash \top =_o \top \longrightarrow f\ \top\ \top =_o g\ \top\ \top}\,\forall\text{E}^+ \quad \dfrac{}{\vdash \top =_o \top}\,\text{refl}}{f \doteq_{o \Rightarrow o \Rightarrow o} g \vdash f\ \top\ \top =_o g\ \top\ \top}\,\rightarrow\text{E}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{f \doteq_{o \Rightarrow o \Rightarrow o} g \vdash (\lambda f.\ f\ \top\ \top)\ f =_o (\lambda f.\ f\ \top\ \top)\ g}{\vdash f \doteq_{o \Rightarrow o \Rightarrow o} g \longrightarrow (\lambda f.\ f\ \top\ \top)\ f =_o (\lambda f.\ f\ \top\ \top)\ g}\,\rightarrow\text{I}}{\vdash \forall f\ g.\ f \doteq_{o \Rightarrow o \Rightarrow o} g \longrightarrow (\lambda f.\ f\ \top\ \top)\ f \doteq_{(o \Rightarrow o \Rightarrow o) \Rightarrow o} (\lambda f.\ f\ \top\ \top)\ g}\,\forall\text{I}}{(\lambda f.\ f\ \top\ \top) \doteq_{(o \Rightarrow o \Rightarrow o) \Rightarrow o} (\lambda f.\ f\ \top\ \top)}\,\text{defn}^\doteq}{\vdots\,2.1}\,(\text{fresh } f, g)$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{p^\bullet \wedge q^\bullet \vdash p^\bullet \wedge q^\bullet}\,\text{assm}}{p^\bullet \wedge q^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet) =_o (\lambda f.\ f\ \top\ \top)}\,\text{defn}^\wedge + \text{beta} \quad (\lambda f.\ f\ \top\ \top) \doteq_{(o \Rightarrow o \Rightarrow o) \Rightarrow o} (\lambda f.\ f\ \top\ \top)\,\text{rewrite}}{p^\bullet \wedge q^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet) \doteq_{(o \Rightarrow o \Rightarrow o) \Rightarrow o} (\lambda f.\ f\ \top\ \top)}\,\text{beta}}{p^\bullet \wedge q^\bullet \vdash (\lambda p\ q.\ (\lambda f.\ f\ p\ q) \doteq_{(o \Rightarrow o \Rightarrow o) \Rightarrow o} (\lambda f.\ f\ \top\ \top))\ p^\bullet\ q^\bullet}\,\text{defn}^\bullet}{p^\bullet \wedge q^\bullet \vdash (\lambda p\ q.\ (\lambda f.\ f\ p\ q) = (\lambda f.\ f\ \top\ \top))^\bullet\ p^\bullet\ q^\bullet}\,\text{defn}^\wedge$$

$$\vdots\,2$$

$$\dfrac{\dfrac{\vdots\,1}{(p \wedge q)^\bullet \vdash (\lambda f.\ f\ p^\bullet\ q^\bullet) =_o (\lambda f.\ f\ \top\ \top)}\,\text{defn}^\wedge + \text{beta} \quad p^\bullet \wedge q^\bullet \vdash (p \wedge q)^\bullet}{(p \wedge q)^\bullet \vdash p^\bullet \wedge q^\bullet}\,\vdots\,2 \quad \text{deduct-antisym}$$

$$\vdash (p \wedge q)^\bullet =_o p^\bullet \wedge q^\bullet$$

Implication

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash [=]_o = [=]_o}\,\text{refl} \quad \dfrac{}{\vdash (p \wedge q)^\bullet =_o p^\bullet \wedge q^\bullet}\,\wedge\bullet \quad \dfrac{}{\vdash p = p}\,\text{refl}}{\vdash ((p \wedge q)^\bullet =_o p) =_o ((p^\bullet \wedge q^\bullet) =_o p^\bullet)}\,\text{mk-comb}^+}{\vdash ((p \wedge q)^\bullet =_o p^\bullet) =_o ((p^\bullet \wedge q^\bullet) =_o p^\bullet)}\,\text{defn}\dot{=}}{\vdash ((p \wedge q)^\bullet \mathrel{\dot{=}}_o p^\bullet) =_o ((p^\bullet \wedge q^\bullet) =_o p^\bullet)}\,\text{defn}\bullet}{\vdash (p \longrightarrow q)^\bullet =_o p^\bullet \longrightarrow q^\bullet}\,\text{defn}\to$$

Universal Quantification

Note that $[\forall]$ is a definition in $\text{PHOL}_{\xi\eta}$, but a primitive symbol in PHOL.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{([\forall]\ (\lambda x.\ p))^\bullet \vdash ([\forall]\ (\lambda x.\ p))^\bullet}\,\text{assm}}{([\forall]\ (\lambda x.\ p))^\bullet \vdash p^\bullet \mathrel{\dot{=}}_{\tau\Rightarrow o} (\lambda x.\ \top^\bullet)}\,\text{defn}\forall + \text{defn}\bullet}{([\forall]\ (\lambda x.\ p))^\bullet \vdash \forall x\ y.\ x \mathrel{\dot{=}}_\tau y \longrightarrow (\lambda x.\ p^\bullet)\ x =_o (\lambda x.\ \top)\ y}\,\text{defn}\dot{=}}{([\forall]\ (\lambda x.\ p))^\bullet \vdash x \mathrel{\dot{=}}_\tau x \longrightarrow (\lambda x.\ p^\bullet)\ x =_o (\lambda x.\ \top)\ x}\,\forall\text{E + beta}}{([\forall]\ (\lambda x.\ p))^\bullet, x \mathrel{\dot{=}}_\tau x \vdash (\lambda x.\ p^\bullet)\ x =_o (\lambda x.\ \top)\ x}\,{\to}\text{E + assm}}{([\forall]\ (\lambda x.\ p))^\bullet, x \mathrel{\dot{=}}_\tau x \vdash (\lambda x.\ p^\bullet)\ x =_o \top}\,\text{beta}}{([\forall]\ (\lambda x.\ p))^\bullet, x \mathrel{\dot{=}}_\tau x \vdash \top =_o (\lambda x.\ p^\bullet)\ x}\,\text{sym} \qquad \dfrac{}{\vdash \top}\,\top\text{I}}{([\forall]\ (\lambda x.\ p))^\bullet, x \mathrel{\dot{=}}_\tau x \vdash (\lambda x.\ p^\bullet)\ x}\,\text{eq-mp}}{([\forall]\ (\lambda x.\ p))^\bullet, x \mathrel{\dot{=}}_\tau x \vdash p^\bullet}\,\text{beta}}{([\forall]\ (\lambda x.\ p))^\bullet \vdash x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet}\,{\to}\text{I}}{([\forall]\ (\lambda x.\ p))^\bullet \vdash \forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet}\,\forall\text{I}^+$$

$[x/x, x/y]$ (x fresh)

$$\vdots\,1$$

Note that this proof uses the *trans*$\dot{=}$ lemma, which is proven in Subsection C.1.6. The proof of that lemma does not rely on the translation of any connectives, and so is safe to use here.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \top}\,\top\text{I}}{(\lambda x.\ p^\bullet)\ x \vdash \top}\,\text{weaken} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{(\forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet) \vdash (\forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet)}\,\text{assm}}{(\forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet) \vdash x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet}\,\forall\text{E}^+ \qquad \dfrac{\dfrac{}{x \mathrel{\dot{=}}_\tau y \vdash x \mathrel{\dot{=}}_\tau y}\,\text{assm}}{x \mathrel{\dot{=}}_\tau y, x \mathrel{\dot{=}}_\tau y \vdash x \mathrel{\dot{=}}_\tau x}\,\text{contract}}{x \mathrel{\dot{=}}_\tau y \vdash x \mathrel{\dot{=}}_\tau x}\,{\to}\text{E}}{(\forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet), x \mathrel{\dot{=}}_\tau y \vdash p^\bullet}\,\top\text{E}}{(\forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet), x \mathrel{\dot{=}}_\tau y, \top \vdash p^\bullet}\,\text{deduct-antisym}}{(\forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet), x \mathrel{\dot{=}}_\tau y \vdash p^\bullet =_o \top}\,\text{beta}}{(\forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet), x \mathrel{\dot{=}}_\tau y \vdash (\lambda x.\ p^\bullet)\ x =_o (\lambda x.\ \top)\ y}\,\text{weaken}}{(\forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet), x \mathrel{\dot{=}}_\tau y \vdash (\lambda x.\ p^\bullet)\ x =_o (\lambda x.\ \top)\ y}\,{\to}\text{I}}{(\forall x.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet) \vdash x \mathrel{\dot{=}}_\tau y \longrightarrow (\lambda x.\ p^\bullet)\ x =_o (\lambda x.\ \top)\ y}\,\forall\text{I}^+}{(\forall x\ y.\ x \mathrel{\dot{=}}_\tau x \longrightarrow p^\bullet) \vdash ((\lambda x.\ p^\bullet) \mathrel{\dot{=}}_{\tau\Rightarrow o} (\lambda x.\ \top))}\,\text{defn}\dot{=}$$

$[x/x, y/y]$ (x, y fresh)

$$\vdots\,2$$

$$\frac{\begin{array}{c}\vdots\,1\\ ([\forall]\ (\lambda x.\ p))^\bullet \vdash \forall x\ y.\, x \doteq_\tau y \longrightarrow p^\bullet\end{array} \qquad \begin{array}{c}\vdots\,2\\ (\forall x.\, x \doteq_\tau x \longrightarrow p^\bullet), x \doteq_\tau y, \top \vdash p^\bullet\end{array}}{\vdash ((\lambda x.\ p^\bullet) \doteq_{o\Rightarrow o} (\lambda x.\ \top)) =_o (\forall x\ y.\, x \doteq_\tau x \longrightarrow p^\bullet)}\ \text{deduct-antisym}$$

$$\frac{\vdash ((\lambda x.\ p^\bullet) \doteq_{o\Rightarrow o} (\lambda x.\ \top)) =_o (\forall x\ y.\, x \doteq_\tau x \longrightarrow p^\bullet)}{\vdash (\lambda z.\ z =_{o\Rightarrow o} (\lambda x.\ \top))\ (\lambda x.\ p^\bullet) =_o (\forall x\ y.\, x \doteq_\tau x \longrightarrow p^\bullet)}\ \text{beta}$$

$$\frac{\vdash (\lambda z.\ z =_{o\Rightarrow o} (\lambda x.\ \top))\ (\lambda x.\ p^\bullet) =_o (\forall x\ y.\, x \doteq_\tau x \longrightarrow p^\bullet)}{\vdash \big((\lambda z.\ z =_{o\Rightarrow o} (\lambda x.\ \top))\ (\lambda x.\ p)\big)^\bullet =_o (\forall x\ y.\, x \doteq_\tau x \longrightarrow p^\bullet)}\ \text{defn}\bullet + \top\bullet$$

$$\frac{\vdash \big((\lambda z.\ z =_{o\Rightarrow o} (\lambda x.\ \top))\ (\lambda x.\ p)\big)^\bullet =_o (\forall x\ y.\, x \doteq_\tau x \longrightarrow p^\bullet)}{\vdash ([\forall]\ (\lambda x.\ p))^\bullet =_o (\forall x.\, x \doteq_\tau x \longrightarrow p^\bullet)}\ \text{defn}\forall$$

Existential Quantification

$$\frac{}{([\exists]\ p)^\bullet \vdash ([\exists]\ p)^\bullet}\ \text{assm}$$

$$\frac{([\exists]\ p)^\bullet \vdash ([\exists]\ p)^\bullet}{([\exists]\ p)^\bullet \vdash \big(\forall z_o.(\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o\big)^\bullet}\ \text{defn}\exists + \text{beta}$$

$$\frac{([\exists]\ p)^\bullet \vdash \big(\forall z_o.(\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o\big)^\bullet}{([\exists]\ p)^\bullet \vdash \forall z.\, z \doteq_o z \longrightarrow ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}\ \forall\bullet + \text{beta}$$

$$[z/z]\ \frac{([\exists]\ p)^\bullet \vdash \forall z.\, z \doteq_o z \longrightarrow ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}{([\exists]\ p)^\bullet \vdash z \doteq_o z \longrightarrow ((\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o)^\bullet}\ \forall\text{E}$$

$$\frac{}{\vdash z =_o z}\ \text{refl} \qquad \frac{\vdash z =_o z}{\vdash z \doteq_o z}\ \text{defn}\doteq$$

$$\frac{([\exists]\ p)^\bullet \vdash z \doteq_o z \longrightarrow ((\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o)^\bullet \qquad \vdash z \doteq_o z}{([\exists]\ p)^\bullet \vdash ((\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o)^\bullet}\ \longrightarrow\text{E}$$

$$\frac{([\exists]\ p)^\bullet \vdash ((\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o)^\bullet}{([\exists]\ p)^\bullet \vdash (\forall x.\, p\ x \longrightarrow z_o)^\bullet \longrightarrow z_o}\ \to\bullet + \text{defn}\bullet$$

$$\frac{}{\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o \vdash \forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o}\ \text{assm}$$

$$\frac{\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o \vdash \forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o}{\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o \vdash (\forall x.\, p\ x \longrightarrow z_o)^\bullet}\ \forall\bullet + \to\bullet$$

$$\frac{([\exists]\ p)^\bullet \vdash (\forall x.\, p\ x \longrightarrow z_o)^\bullet \longrightarrow z_o \qquad \forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o \vdash (\forall x.\, p\ x \longrightarrow z_o)^\bullet}{([\exists]\ p)^\bullet, \forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o \vdash z_o}\ \longrightarrow\text{E}$$

$$\frac{([\exists]\ p)^\bullet, \forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o \vdash z_o}{([\exists]\ p)^\bullet \vdash (\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o) \longrightarrow z_o}\ \longrightarrow\text{I}$$

$$(z_o\ \text{fresh})\ \frac{([\exists]\ p)^\bullet \vdash (\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o) \longrightarrow z_o}{([\exists]\ p)^\bullet \vdash \forall z_o.(\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o) \longrightarrow z_o}\ \forall\text{I}$$

$$\frac{([\exists]\ p)^\bullet \vdash \forall z_o.(\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o) \longrightarrow z_o}{([\exists]\ p)^\bullet \vdash [\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x)}\ \text{defn}\exists$$

$$\vdots\,1$$

$$\frac{}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash [\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x)}\ \text{assm}$$

$$\frac{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash [\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x)}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash \forall z_o.(\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o) \longrightarrow z_o}\ \text{defn}\exists$$

$$[z/z]\ \frac{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash \forall z_o.(\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z_o) \longrightarrow z_o}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash (\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z) \longrightarrow z}\ \forall\text{E}$$

$$\frac{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash (\forall x.(x \doteq_\tau x \longrightarrow p^\bullet\ x) \longrightarrow z) \longrightarrow z}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash (\forall x.\, p\ x \longrightarrow z)^\bullet \longrightarrow z}\ \forall\bullet + \to\bullet + \text{defn}\bullet$$

$$\frac{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash (\forall x.\, p\ x \longrightarrow z)^\bullet \longrightarrow z}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}\ \to\bullet + \text{defn}\bullet$$

$$\frac{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x), z \doteq_o z \vdash ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}\ \text{weaken}$$

$$\frac{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x), z \doteq_o z \vdash ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash z \doteq_o z \longrightarrow ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}\ \longrightarrow\text{I}$$

$$(z\ \text{fresh})\ \frac{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash z \doteq_o z \longrightarrow ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash \forall z.\, z \doteq_o z \longrightarrow ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}\ \forall\text{I}$$

$$\frac{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash \forall z.\, z \doteq_o z \longrightarrow ((\forall x.\, p\ x \longrightarrow z) \longrightarrow z)^\bullet}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash \big(\forall z_o.(\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o\big)^\bullet}\ \forall\bullet + \text{beta}$$

$$\frac{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash \big(\forall z_o.(\forall x.\, p\ x \longrightarrow z_o) \longrightarrow z_o\big)^\bullet}{[\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash ([\exists]\ p)^\bullet}\ \text{defn}\exists + \text{beta}$$

$$\frac{\begin{array}{c}\vdots\,1\\ ([\exists]\ p)^\bullet \vdash [\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x)\end{array} \qquad [\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x) \vdash ([\exists]\ p)^\bullet}{\vdash ([\exists]\ p)^\bullet =_o [\exists]\ (\lambda x_\tau.\ x \doteq_\tau x \longrightarrow p^\bullet\ x)}\ \text{defn}\exists$$

Disjunction

$$\vdash (p \lor q)^\bullet =_o p^\bullet \lor q^\bullet$$

The proof follows by defn$\lor$ and $\to\bullet$, in a similar fashion to the proof of $\exists\bullet$.

False

$$\vdash \bot^\bullet =_o \bot$$

The proof follows by defn$\bot$, $\forall\bullet$, and *refl*.

Negation

$$\vdash (\neg p)^\bullet =_o \neg p^\bullet$$

The proof follows by defn$\neg$, $\bot\bullet$, and $\to\bullet$, in a similar fashion to the proof of $\to\bullet$.

Universal Type-Quantification

Note that $[\forall_{\mathsf{ty}}]$ is a definition in $\mathrm{PHOL}_{\xi\eta}$, but a primitive symbol in PHOL.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet \vdash ([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet}\ \text{assm}}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet \vdash p^\bullet \doteq_{\Pi\alpha.\ o} (\Lambda\alpha.\ \top^\bullet)}\ \text{defn}\forall + \text{defn}\bullet}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet \vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ p^\bullet)\ [:\alpha:] =_o (\Lambda\alpha.\ \top)\ [:\alpha:]}\ \text{defn}\doteq}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet \vdash \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ p^\bullet)\ [:\alpha:] =_o (\Lambda\alpha.\ \top)\ [:\alpha:]}\ \forall\text{E} + \text{beta}\quad \dfrac{}{\mathcal{E}(\alpha) \vdash \mathcal{E}(\alpha)}\ \text{assm}}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet, \mathcal{E}(\alpha) \vdash (\Lambda\alpha.\ p^\bullet)\ [:\alpha:] =_o (\Lambda\alpha.\ \top)\ [:\alpha:]}\ \to\text{E}}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet, \mathcal{E}(\alpha) \vdash (\Lambda\alpha.\ p^\bullet)\ [:\alpha:] =_o \top}\ \text{beta}}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet, \mathcal{E}(\alpha) \vdash \top =_o (\Lambda\alpha.\ p^\bullet)\ [:\alpha:]}\ \text{sym}\qquad \dfrac{}{\vdash \top}\ \top\text{I}}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet, \mathcal{E}(\alpha) \vdash (\Lambda\alpha.\ p^\bullet)\ [:\alpha:]}\ \text{eq-mp}}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet, \mathcal{E}(\alpha) \vdash p^\bullet[\alpha/\alpha]}\ \text{beta}}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet, \vdash \mathcal{E}(\alpha)[\alpha/\alpha] \longrightarrow p^\bullet[\alpha/\alpha]}\ \to\text{I}}{([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet \vdash [\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet)}\ \forall_{\mathsf{ty}}\text{I}\ (\alpha\text{ fresh})}{\vdots\ 1}\ [\alpha/\alpha]$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{[\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet) \vdash [\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet)}\ \text{assm}}{[\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet) \vdash (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet)\ [:\alpha:]}\ \forall_{\mathsf{ty}}\text{E}}{[\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet) \vdash \mathcal{E}(\alpha) \longrightarrow p^\bullet}\ \text{tybeta}\qquad \dfrac{}{\mathcal{E}(\alpha) \vdash \mathcal{E}(\alpha)}}{[\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet), \mathcal{E}(\alpha) \vdash p^\bullet}\ \to\text{E}}{p^\bullet \vdash \top}\ \text{weaken}\qquad \dfrac{}{\vdash \top}\ \top\text{I}}{[\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet), \mathcal{E}(\alpha) \vdash p^\bullet =_o \top}\ \text{deduct-antisym}}{[\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet), \mathcal{E}(\alpha) \vdash (\Lambda\alpha.\ p^\bullet)\ [:\alpha:] =_o (\Lambda\alpha.\ \top)\ [:\alpha:]}\ \text{tybeta}}{[\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet), \vdash \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ p^\bullet)\ [:\alpha:] =_o (\Lambda\alpha.\ \top)\ [:\alpha:]}\ \to\text{I}}{[\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet), \vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ p^\bullet)\ [:\alpha:] =_o (\Lambda\alpha.\ \top)\ [:\alpha:]}\ \forall\text{I}^+\ (\alpha\text{ fresh})}{[\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet), \vdash ((\Lambda\alpha.\ p^\bullet) \doteq_{\Pi\alpha.\ o} (\Lambda\alpha.\ \top))}\ \text{defn}\doteq}{\vdash ((\Lambda\alpha.\ p^\bullet) \doteq_{o\Rightarrow o} (\Lambda\alpha.\ \top)) =_o [\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet)}\ \text{deduct-antisym}}{\vdash (\lambda z.\ z =_{o\Rightarrow o} (\Lambda\alpha.\ \top))\ (\Lambda\alpha.\ p^\bullet) =_o [\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet)}\ \text{beta}}{\vdash \big((\lambda z.\ z =_{o\Rightarrow o} (\Lambda\alpha.\ \top))\ (\Lambda\alpha.\ p)\big)^\bullet =_o [\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p)^\bullet}\ \text{defn}\bullet + \top\bullet}{\vdash ([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet =_o [\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p)^\bullet}\ \text{defn}\forall$$

$$([\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ p))^\bullet, \mathcal{E}(\alpha) \vdash [\forall_{\mathsf{ty}}]\ (\Lambda\alpha.\ \mathcal{E}(\alpha) \longrightarrow p^\bullet)\ \vdots\ 1$$

B.2.10 mk-comb$\doteq$

The rule

$$\frac{\Gamma \vdash s \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} t \qquad \Delta \vdash u \mathrel{\dot{=}}_{\tau_1} v}{\Gamma, \Delta \vdash s\ u \mathrel{\dot{=}}_{\tau_2} t\ v}\ \text{mk-comb}\dot{=}$$

holds in PHOL.

Proof.

$$[u/x, v/y]\ \frac{\dfrac{\dfrac{\Gamma \vdash s \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} t}{\Gamma \vdash \forall x\ y.\ x \mathrel{\dot{=}}_{\tau_1} y \longrightarrow s\ x \mathrel{\dot{=}}_{\tau_2} t\ y}\ \text{defn}}{\Gamma \vdash u \mathrel{\dot{=}}_{\tau_1} v \longrightarrow s\ u \mathrel{\dot{=}}_{\tau_2} t\ v}\ (\forall\text{E})^+ \qquad \Delta \vdash u \mathrel{\dot{=}}_{\tau_1} v}{\Gamma, \Delta \vdash s\ u \mathrel{\dot{=}}_{\tau_2} t\ v}\ \rightarrow\text{E}$$

□

B.2.11 tyapp$\dot{=}$

The rule

$$\frac{\Gamma \vdash s^\bullet \mathrel{\dot{=}}_{\Lambda\alpha.\ \tau} t^\bullet}{\Gamma, \mathcal{E}(\alpha) \vdash s^\bullet\ [:\alpha:] \mathrel{\dot{=}}_\tau t^\bullet\ [:\alpha:]}\ \text{tyapp}\dot{=}$$

holds in PHOL.

Proof.

$$[\alpha/\alpha]\ \frac{\dfrac{\dfrac{\Gamma \vdash s^\bullet \mathrel{\dot{=}}_{\Lambda\alpha.\ \tau} t^\bullet}{\Gamma \vdash \forall_{\text{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow s^\bullet\ [:\alpha:] \mathrel{\dot{=}}_\tau t^\bullet\ [:\alpha:]}\ \text{defn}\dot{=}}{\Gamma \vdash \mathcal{E}(\alpha) \longrightarrow s^\bullet\ [:\alpha:] \mathrel{\dot{=}}_\tau t^\bullet\ [:\alpha:]}\ \forall_{\text{ty}}\text{E} \qquad \dfrac{}{\mathcal{E}(\alpha) \vdash \mathcal{E}(\alpha)}\ \text{assm}}{\Gamma, \mathcal{E}(\alpha) \vdash s^\bullet\ [:\alpha:] \mathrel{\dot{=}}_\tau t^\bullet\ [:\alpha:]}\ \rightarrow\text{E}$$

□

B.2.12 The Substitution Lemma

The substitution lemma says that, assuming that between two substitutions θ_l and θ_r the substitute term for each variable is extensionally partially-equivalent to the other, then substituting into a term s by θ_l in extensionally partially-equivalent to substituting by θ_r.

Stating this formally, we first define $\mathfrak{R}$ as

$$\mathfrak{R}(s, \theta_l, \theta_r) \triangleq \{\!\{x[\theta_l] \mathrel{\dot{=}} x[\theta_r] \mid x \in \text{FV}(s)\}\!\}$$

Note that this devolves to $\mathcal{R}(s)$ when the substitution is empty.

The main theorem is as follows: If s does not contain $[=]$, except on o and ι, then

$$\frac{}{\mathcal{E}(s), \mathfrak{R}(s, \theta_l, \theta_r) \vdash s[\theta_l] \doteq_\tau s[\theta_r]}\ \text{subst. lemma.}$$

Proof. By induction on s.

Case x:

$$\dfrac{\dfrac{\dfrac{}{x[\theta_l] \doteq_\tau x[\theta_r] \vdash x[\theta_l] \doteq_\tau x[\theta_r]}\ \text{assm}}{\mathfrak{R}(x, \theta_l, \theta_r) \vdash x[\theta_l] \doteq_\tau x[\theta_r]}\ \text{defn}\mathfrak{R}}{\mathcal{E}(x), \mathfrak{R}(x, \theta_l, \theta_r) \vdash x[\theta_l] \doteq_\tau x[\theta_r]}\ \text{defn}\mathcal{E}$$

Case $\lambda z.\ s$:

To begin, there are four subcases, depending on whether z is in $\mathrm{Vars}(\theta_l)$ or $\mathrm{Vars}(\theta_r)$.

If a/z is in θ_l, we rebind θ_l as $(\theta_l, a/z) \coloneqq \theta_l$. Then $(\lambda z.\ s)[\theta_l, a/z] \equiv (\lambda z.\ s[\theta_l])$ by the definition of substitution; if b/z is in θ_r, then we perform a similar step for θ_r. In all cases, by this rewriting, the goal is then

$$\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r) \vdash (\lambda z.\ s[\theta_l]) \doteq_{\tau_1 \Rightarrow \tau_2} (\lambda z.\ s[\theta_r])$$

$$(x, y \text{ fresh})\ \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{E}(s), \mathfrak{R}(s, (\theta_l, x/z), (\theta_r, y/z)) \vdash s[\theta_l, x/z] \doteq_{\tau_2} s[\theta_r, y/z]}\ \text{I.H.}}{\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r), z[\theta_l, x/z] \doteq_{\tau_1} z[\theta_l, y/z] \vdash s[\theta_l, x/z] \doteq_{\tau_2} s[\theta_r, y/z]}\ \text{defn}\mathfrak{R}}{\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r), x \doteq_{\tau_1} y \vdash s[\theta_l, x/z] \doteq_{\tau_2} s[\theta_r, y/z]}\ \text{defn-subst}}{\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r), x \doteq_{\tau_1} y \vdash (\lambda z.\ s[\theta_l])\ x \doteq_{\tau_2} (\lambda z.\ s[\theta_r])\ y}\ \text{beta}}{\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r) \vdash x \doteq_{\tau_1} y \longrightarrow (\lambda z.\ s[\theta_l])\ x \doteq_{\tau_2} (\lambda z.\ s[\theta_r])\ y}\ \to\text{I}}{\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r) \vdash \forall x\ y.\ x \doteq_{\tau_1} y \longrightarrow (\lambda z.\ s[\theta_l])\ x \doteq_{\tau_2} (\lambda z.\ s[\theta_r])\ y}\ \forall\text{I}^+}{\mathcal{E}(s), \mathfrak{R}(\lambda z.\ s, \theta_l, \theta_r) \vdash (\lambda z.\ s[\theta_l]) \doteq_{\tau_1 \Rightarrow \tau_2} (\lambda z.\ s[\theta_r])}\ \text{defn}\doteq$$

Case $s\ t$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{E}(s), \mathfrak{R}(s, \theta_l, \theta_r) \vdash s[\theta_l] \doteq_{\sigma \Rightarrow \tau} s[\theta_r]}\ \text{I.H.} \quad \dfrac{}{\mathcal{E}(t), \mathfrak{R}(t, \theta_l, \theta_r) \vdash t[\theta_l] \doteq_\sigma t[\theta_r]}\ \text{I.H.}}{\mathcal{E}(s), \mathcal{E}(t), \mathfrak{R}(s, \theta_l, \theta_r), \mathfrak{R}(t, \theta_l, \theta_r) \vdash s[\theta_l]\ t[\theta_l] \doteq_\tau s[\theta_r]\ t[\theta_r]}\ \text{mk-comb}\doteq}{\mathcal{E}(s), \mathcal{E}(t), \mathfrak{R}(s\ t, \theta_l, \theta_r) \vdash (s\ t)[\theta_l] \doteq_\tau (s\ t)[\theta_r]}\ \text{defn}\mathfrak{R}}{\mathcal{E}(s), \mathcal{E}(t), \mathfrak{R}(s\ t, \theta_l, \theta_r) \vdash (s\ t)[\theta_l] \doteq_\tau (s\ t)[\theta_r]}\ \text{contract}}{\mathcal{E}(s\ t), \mathfrak{R}(s\ t, \theta_l, \theta_r) \vdash (s\ t)[\theta_l] \doteq_\tau (s\ t)[\theta_r]}\ \text{defn}\mathcal{E}$$

Case $[=]_\chi$:

Only occurs as $[=]_o$ or $[=]_\iota$. Let χ stand for the type in either case; then:

$$\dfrac{\dfrac{\dfrac{}{x_1 =_\chi y_1 \vdash x_1 =_\chi y_1}\,\text{assm}}{x_1 =_\chi y_1 \vdash y_1 =_\chi x_1}\,\text{sym} \quad \dfrac{}{x_1 =_\chi x_2 \vdash x_1 =_\chi x_2}\,\text{assm} \quad \dfrac{}{x_2 =_\chi y_2 \vdash x_2 =_\chi y_2}\,\text{assm}}{x_1 =_\chi y_1, x_2 =_\chi y_2, x_1 =_\chi x_2 \vdash y_1 =_\chi y_2}\,\text{trans}^+$$

$$\vdots\,1$$

$$\dfrac{\dfrac{}{x_1 =_\chi y_1 \vdash x_1 =_\chi y_1}\,\text{assm} \quad \dfrac{}{y_1 =_\chi y_2 \vdash y_1 =_\chi y_2}\,\text{assm} \quad \dfrac{\dfrac{}{x_2 =_\chi y_2 \vdash y_2 =_\chi x_2}\,\text{assm}}{x_2 =_\chi y_2 \vdash y_2 =_\chi x_2}\,\text{sym}}{x_1 =_\chi y_1, x_2 =_\chi y_2, y_1 =_\chi y_2 \vdash x_1 =_\chi x_2}\,\text{trans}^+$$

$$\vdots\,2$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdots\,1 \quad \vdots\,2}{x_1 =_\chi y_1, x_2 =_\chi y_2, x_1 =_\chi x_2 \vdash y_1 =_\chi y_2 \quad x_1 =_\chi y_1, x_2 =_\chi y_2, y_1 =_\chi y_2 \vdash x_1 =_\chi x_2}\,\text{deduct-antisym}}{x_1 =_\chi y_1, x_2 =_\chi y_2, x_1 =_\chi x_2, x_1 =_\chi y_1, x_2 =_\chi y_2 \vdash (x_1 =_\chi x_2) =_o (y_1 =_\chi y_2)}\,\text{contract}}{x_1 =_\chi y_1, x_2 =_\chi y_2 \vdash (x_1 =_\chi x_2) =_o (y_1 =_\chi y_2)}\,\forall\text{I}^+ + \to\text{I}}{\vdash \forall x_1\ y_1.\, x_1 =_\chi y_1 \longrightarrow (\forall x_2\ y_2.\, x_2 =_\chi y_2 \longrightarrow (x_1 =_\chi x_2) =_o (y_1 =_\chi y_2))}\,\text{defn}\dot{=}^+}{\vdash [=]_\chi \doteq_{\chi\Rightarrow\chi\Rightarrow o} [=]_\chi}\,\text{defn-subst}}{\vdash [=]_\chi[\theta_l] \doteq_{\chi\Rightarrow\chi\Rightarrow o} [=]_\chi[\theta_r]}\,\text{defn}\mathfrak{R}}{\mathfrak{R}([=], \theta_l, \theta_r) \vdash [=]_\chi[\theta_l] \doteq_{\chi\Rightarrow\chi\Rightarrow o} [=]_\chi[\theta_r]}\,\text{defn}\mathcal{E}}{\mathcal{E}([=]), \mathfrak{R}([=], \theta_l, \theta_r) \vdash [=]_\chi[\theta_l] \doteq_{\chi\Rightarrow\chi\Rightarrow o} [=]_\chi[\theta_r]}\ (\text{fresh } x_1, y_1)$$

Case $[\dot{=}]_\alpha$:

$$\dfrac{\dfrac{\dfrac{}{\mathcal{E}([\dot{=}]_\alpha) \vdash [\dot{=}]_\alpha \doteq_{\alpha\Rightarrow\alpha\Rightarrow o} [\dot{=}]_\alpha}\,\text{defn}\mathcal{E} + \wedge\text{E}}{\mathcal{E}([\dot{=}]_\alpha) \vdash [\dot{=}]_\alpha[\theta_l] \doteq_{\alpha\Rightarrow\alpha\Rightarrow o} [\dot{=}]_\alpha[\theta_r]}\,\text{defn-subst}}{\mathcal{E}([\dot{=}]_\alpha), \mathfrak{R}([\dot{=}], \theta_l, \theta_r) \vdash [\dot{=}]_\alpha[\theta_l] \doteq_{\alpha\Rightarrow\alpha\Rightarrow o} [\dot{=}]_\alpha[\theta_r]}\,\text{defn}\mathfrak{R}$$

Case $\Lambda\alpha.\ s$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{E}(s) \vdash s[\theta_l] \doteq_\tau s[\theta_l]}\,\text{I.H.}}{\mathcal{E}(\Lambda\alpha.\ s), \mathcal{E}(\alpha) \vdash s[\theta_l] \doteq_\tau s[\theta_l]}\,\text{defn}\mathcal{E} + \text{weaken}^*}{\mathcal{E}(\Lambda\alpha.\ s), \mathcal{E}(\alpha), \Gamma \vdash s[\theta_l][\alpha/\alpha] \doteq_\tau s[\theta_r][\alpha/\alpha]}\,\text{defn-subst}}{\mathcal{E}(\Lambda\alpha.\ s), \mathcal{E}(\alpha), \Gamma \vdash (\Lambda\alpha.\ s[\theta_l])\ [:\alpha:] \doteq_\tau (\Lambda\alpha.\ s[\theta_r])\ [:\alpha:]}\,\text{tybeta}}{\mathcal{E}(\Lambda\alpha.\ s), \Gamma \vdash \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ s[\theta_l])\ [:\alpha:] \doteq_\tau (\Lambda\alpha.\ s[\theta_r])\ [:\alpha:]}\,\to\text{I}}{\mathcal{E}(\Lambda\alpha.\ s), \Gamma \vdash \forall_{\text{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ s[\theta_l])\ [:\alpha:] \doteq_\tau (\Lambda\alpha.\ s[\theta_r])\ [:\alpha:]}\,\forall_{\text{ty}}\text{I}}{\mathcal{E}(\Lambda\alpha.\ s), \Gamma \vdash (\Lambda\alpha.\ s[\theta_l]) \doteq_{\Pi\alpha.\,\tau} (\Lambda\alpha.\ s[\theta_r])}\,\text{defn}\dot{=}}{\mathcal{E}(\Lambda\alpha.\ s), \Gamma \vdash (\Lambda\alpha.\ s)[\theta_l] \doteq_{\Pi\alpha.\,\tau} (\Lambda\alpha.\ s)[\theta_r]}\,\text{defn-subst}$$

(α fresh)

(renaming α if necessary)

Case $s\ [:\sigma:]$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{E}(s), \mathfrak{R}(s, \theta_l, \theta_r) \vdash s[\theta_l] \stackrel{\bullet}{=}_{\Pi\alpha.\ \tau} s[\theta_r]}\ \text{I.H.}}{\mathcal{E}(s), \mathfrak{R}(s, \theta_l, \theta_r) \vdash s[\theta_l]\ [:\sigma:] \stackrel{\bullet}{=}_{\tau} s[\theta_r]\ [:\sigma:]}\ \text{tyapp}\stackrel{\bullet}{=}}{\mathcal{E}(s), \mathfrak{R}(s, \theta_l, \theta_r) \vdash (s\ [:\sigma:])[\theta_l] \stackrel{\bullet}{=}_{\tau} (s\ [:\sigma:])[\theta_r]}\ \text{defn-subst}}{\mathcal{E}(s), \mathfrak{R}(s\ [:\sigma:], \theta_l, \theta_r) \vdash (s\ [:\sigma:])[\theta_l] \stackrel{\bullet}{=}_{\tau} (s\ [:\sigma:])[\theta_r]}\ \text{defn}\mathfrak{R}}{\mathcal{E}(s\ [:\sigma:]), \mathfrak{R}(s\ [:\sigma:], \theta_l, \theta_r) \vdash (s\ [:\sigma:])[\theta_l] \stackrel{\bullet}{=}_{\tau} (s\ [:\sigma:])[\theta_r]}\ \text{defn}\mathcal{E} + \text{weaken}^*$$

□

Appendix C

C.1 Proofs of Translation

C.1.1 Translation Validity

Recall, the proof rules of $\text{PHOL}_{\xi\eta}$ are

Structural

$$\frac{}{p \vdash p}\ \text{assm}$$

$$\frac{\Gamma \vdash p}{\Gamma, \Delta \vdash p}\ \text{weaken} \qquad \frac{\Gamma, \Delta, \Delta \vdash p}{\Gamma, \Delta \vdash p}\ \text{contract}$$

Equality

$$\frac{}{\vdash s = s}\ \text{refl} \qquad \frac{\Gamma \vdash s = t \quad \Delta \vdash t = u}{\Gamma, \Delta \vdash s = u}\ \text{trans}$$

$$\frac{\Gamma \vdash s = t \quad \Delta \vdash u = v}{\Gamma, \Delta \vdash s\ u = t\ v}\ \text{mk-comb} \qquad \frac{}{\vdash (\lambda x.\ s)\ t = s[t/x]}\ \text{beta}$$

$$(x \text{ not in } \Gamma)\ \frac{\Gamma \vdash s = t}{\Gamma \vdash (\lambda x.\ s) = (\lambda x.\ t)}\ \text{abs} \qquad (x \text{ not in } t)\ \frac{}{\vdash (\lambda x.\ t\ x) = t}\ \text{eta}$$

$$\frac{\Gamma \vdash s = t}{\Gamma \vdash s\ [{:}\tau{:}] = t\ [{:}\tau{:}]}\ \text{tyapp} \qquad \frac{}{\vdash (\Lambda\alpha.\ s)\ [{:}\sigma{:}] = s[\sigma/\alpha]}\ \text{tybeta}$$

$$(\alpha \text{ not in } \Gamma)\ \frac{\Gamma \vdash s = t}{\Gamma \vdash (\Lambda\alpha.\ s) = (\Lambda\alpha.\ t)}\ \text{tyabs} \qquad (\alpha \text{ not in } f)\ \frac{}{\vdash (\Lambda\alpha.\ f\ [{:}\alpha{:}]) = f}\ \text{tyeta}$$

$$\frac{\Gamma \vdash s =_o t \quad \Delta \vdash s}{\Gamma, \Delta \vdash t}\ \text{eq-mp} \qquad \frac{\Gamma, q \vdash p \quad \Delta, p \vdash q}{\Gamma, \Delta \vdash p =_o q}\ \text{deduct-antisym}$$

Axioms

$$(x \text{ not in } p)\ \frac{}{\vdash p\ s \longrightarrow p\ (\varepsilon x.\ p\ x)}\ \text{select-ax} \qquad \frac{}{\vdash \exists (f : \iota \Rightarrow \iota).\ \text{inj}\ f \wedge \neg\, \text{onto}\ f}\ \text{infinity-ax}$$

Proof of correctness of translation of theorems is shown by induction on the proof tree of

the theorem.

C.1.2 assm

The rule *assm* is

$$\frac{}{p \vdash p}\ \text{assm}$$

and translated, this rule is

$$\frac{}{\mathcal{E}(p^\bullet), \mathcal{R}(p^\bullet), p^\bullet \vdash p^\bullet}\ \text{assm}\bullet$$

This rule is derivable as follows

$$\frac{\dfrac{}{p^\bullet \vdash p^\bullet}\ \text{assm}}{\mathcal{E}(p^\bullet), \mathcal{R}(p^\bullet), p^\bullet \vdash p^\bullet}\ \text{weaken}$$

C.1.3 weaken

The rule weaken is

$$\frac{\Gamma \vdash p}{\Gamma, \Delta \vdash p}\ \text{weaken}$$

The translated rule is

$$\frac{\mathcal{E}(\Gamma^\bullet, p^\bullet), \mathcal{R}(\Gamma^\bullet, p^\bullet), \Gamma^\bullet \vdash p^\bullet}{\mathcal{E}(\Gamma^\bullet, \Delta^\bullet, p^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, p^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash p^\bullet}$$

which follows immediately from the definition of $\mathcal{E}$, the definition of $reflSetSym$, and *weaken*.

C.1.4 contract

The rule *contract* is

$$\frac{\Gamma, \Delta, \Delta \vdash p}{\Gamma, \Delta \vdash p}\ \text{contract}$$

The translated rule is

$$\frac{\mathcal{E}(\Gamma^\bullet, \Delta^\bullet, \Delta^\bullet, p^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, \Delta^\bullet, p^\bullet), \Gamma^\bullet, \Delta^\bullet, \Delta^\bullet \vdash p^\bullet}{\mathcal{E}(p^\bullet), \mathcal{R}(p^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash p^\bullet}$$

which follows immediately from the definition of $\mathcal{E}$, the definition of $\mathcal{R}$, and *contract*.

C.1.5 refl

The rule refl is

$$\frac{}{\vdash s = s}\ \text{refl}$$

Translated (with slight simplification using *contract*), this rule is

$$\frac{}{\mathcal{E}(\tau), \mathcal{E}(s^\bullet), \mathcal{R}(s^\bullet) \vdash s^\bullet \mathrel{\dot{=}_\tau} s^\bullet}\ \text{refl}\bullet$$

which we can prove admissible by appeal to the substitution lemma on an empty substitution (Subsection B.2.12).

C.1.6 trans

The rule trans is

$$\frac{\Gamma \vdash s = t \qquad \Delta \vdash t = u}{\Gamma, \Delta \vdash s = u}\ \text{trans}$$

translated (using the specialised translation for $=$), this rule is

$$\frac{\mathcal{E}(\tau), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \mathrel{\dot{=}_\tau} t^\bullet \qquad \mathcal{E}(\tau), \mathcal{E}(\Delta^\bullet, t^\bullet, u^\bullet), \mathcal{R}(\Delta^\bullet, t^\bullet, u^\bullet), \Delta^\bullet \vdash t^\bullet \mathrel{\dot{=}_\tau} u^\bullet}{\mathcal{E}(\tau), \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, s^\bullet, u^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, s^\bullet, u^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash s^\bullet \mathrel{\dot{=}_\tau} u^\bullet}\ \text{trans}\bullet$$

To prove this, we instead show the more general rule

$$\frac{\mathcal{E}(\tau), \mathcal{E}(\Gamma, s, t), \mathbf{U}\mathcal{R}_t, \Gamma \vdash s \mathrel{\dot{=}_\tau} t \qquad \mathcal{E}(\tau), \mathcal{E}(\Delta, s, t), \mathbf{U}\mathcal{R}_t, \Delta \vdash t \mathrel{\dot{=}_\tau} u}{\mathcal{E}(\tau), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash s \mathrel{\dot{=}_\tau} u}\ \text{trans}\dot{=}$$

the correctness of *trans*• then follows by careful application of *drop-unused*$\mathcal{R}$ and *drop-unused*$\mathcal{E}$.

The rule *trans*$\dot{=}$ is admissible.

Proof. By induction on τ.

Case χ (o/ι):

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Gamma, s), \mathcal{R}(\Gamma, s), \Gamma \vdash s \mathrel{\dot{=}}_\chi t}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Gamma, s), \mathcal{R}(\Gamma, s), \Gamma \vdash s =_\chi t}\,\text{defn}\dot{=} \quad \dfrac{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Delta, u), \mathcal{R}(\Delta, u), \Delta \vdash t \mathrel{\dot{=}}_\chi u}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Delta, u), \mathcal{R}(\Delta, u), \Delta \vdash t =_\chi u}\,\text{defn}\dot{=}}{\mathcal{E}(t), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Gamma, \Delta, s, u), \mathcal{R}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash s =_\chi u}\,\text{trans}}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Gamma, \Delta, s, u), \mathcal{R}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash s =_\chi u}\,\text{contract}}{\mathcal{E}(t), \mathcal{E}(\mathbf{U}\mathcal{R}_t), \mathcal{E}(\Gamma, \Delta, s, u), \mathcal{R}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash s =_\chi u}\,\text{drop-unused}\mathcal{R}}{\mathbf{U}\mathcal{E}_t, \mathcal{E}(\Gamma, \Delta, s, u), \mathcal{R}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash s =_\chi u}\,\text{contract}}{\mathcal{E}(\Gamma, \Delta, s, u), \mathcal{R}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash s =_\chi u}\,\text{drop-unused}\mathcal{E}}{\mathcal{E}(\Gamma, \Delta, s, u), \mathcal{R}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash s \mathrel{\dot{=}}_\chi u}\,\text{defn}\dot{=}$$

n.b. $\mathcal{E}(\chi) \equiv \emptyset$

Case $\tau_1 \Rightarrow \tau_2$:

$$[x/x, y/y]\ \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, s), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Gamma \vdash s \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} t}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, s), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Gamma \vdash \forall x\ y.\, x \mathrel{\dot{=}}_{\tau_1} y \longrightarrow s\ x \mathrel{\dot{=}}_{\tau_2} t\ y}\,\text{defn}\dot{=}}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, s), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Gamma \vdash x \mathrel{\dot{=}}_{\tau_1} y \longrightarrow s\ x \mathrel{\dot{=}}_{\tau_2} t\ y}\,\forall\text{E + beta}}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, s), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Gamma, x \mathrel{\dot{=}}_{\tau_1} y \vdash s\ x \mathrel{\dot{=}}_{\tau_2} t\ y}\,\rightarrow\text{E + assm}}{\mathcal{E}(\tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, s), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Gamma, x \mathrel{\dot{=}}_{\tau_1} y \vdash s\ x \mathrel{\dot{=}}_{\tau_2} t\ y}\,\text{weaken}$$

$$\vdots\ 2.1$$

$$[y/y_1, y/y_2]\ \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Delta, u), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Delta \vdash t \mathrel{\dot{=}}_{\tau_1 \Rightarrow \tau_2} u}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Delta, u), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Delta \vdash \forall y_1\ y_2.\, y_1 \mathrel{\dot{=}}_{\tau_1} y_2 \longrightarrow t\ y_1 \mathrel{\dot{=}}_{\tau_2} u\ y_2}\,\text{defn}\dot{=}}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Delta, u), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Delta \vdash y \mathrel{\dot{=}}_{\tau_1} y \longrightarrow t\ y \mathrel{\dot{=}}_{\tau_2} u\ y}\,\forall\text{E + beta}}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Delta, u), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Delta, y \mathrel{\dot{=}}_{\tau_1} y \vdash t\ y \mathrel{\dot{=}}_{\tau_2} u\ y}\,\rightarrow\text{E + assm}}{\mathcal{E}(\tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Delta, u), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \Delta, y \mathrel{\dot{=}}_{\tau_1} y \vdash t\ y \mathrel{\dot{=}}_{\tau_2} u\ y}\,\text{weaken}$$

$$\vdots\ 2.2$$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{x \mathrel{\dot{=}}_{\tau_1} y \vdash x \mathrel{\dot{=}}_{\tau_1} y}\,\text{assm}}{\mathcal{E}(\tau_1), x \mathrel{\dot{=}}_{\tau_1} y \vdash y \mathrel{\dot{=}}_{\tau_1} x}\,\text{sym}\dot{=}}{\mathcal{E}(\tau_1), x \mathrel{\dot{=}}_{\tau_1} y \vdash y \mathrel{\dot{=}}_{\tau_1} x}\,\text{weaken} \qquad \dfrac{\dfrac{}{x \mathrel{\dot{=}}_{\tau_1} y \vdash x \mathrel{\dot{=}}_{\tau_1} y}\,\text{assm}}{\mathcal{E}(\tau_1), x \mathrel{\dot{=}}_{\tau_1} y \vdash x \mathrel{\dot{=}}_{\tau_1} y}\,\text{weaken}}{\mathcal{E}(\tau_1), x \mathrel{\dot{=}}_{\tau_1} y, x \mathrel{\dot{=}}_{\tau_1} y \vdash y \mathrel{\dot{=}}_{\tau_1} y}\,\text{I.H}$$

$$\vdots\ 1$$

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\overset{\vdots\,1}{\mathcal{E}(\tau_1), x \doteq_{\tau_1} y, x \doteq_{\tau_1} y \vdash y \doteq_{\tau_1} y} \qquad \dfrac{\overset{\vdots\,2.1}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, s), \Gamma, x \doteq_{\tau_1} y \vdash s\ x \doteq_{\tau_2} t\ y} \quad \overset{\vdots\,2.2}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Delta, u), \Delta, y \doteq_{\tau_1} y \vdash t\ y \doteq_{\tau_2} u\ y}}{\mathcal{E}(\tau_2), \mathcal{E}(\Gamma, \Delta, s, u), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \Gamma, \mathcal{E}(\tau_1 \Rightarrow \tau_2), \Delta, y \doteq_{\tau_1} y, x \doteq_{\tau_1} y \vdash s\ x \doteq_{\tau_2} u\ y}\;\text{I.H.}}{\mathcal{E}(\tau_1), \mathcal{E}(\tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(t), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta, x \doteq_{\tau_1} y, x \doteq_{\tau_1} y, x \doteq_{\tau_1} y \vdash s\ x =_{\tau_2} u\ y}\;\text{ante-subst}}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(t), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta, x \doteq_{\tau_1} y \vdash s\ x =_{\tau_2} u\ y}\;\text{contract + defn}\mathcal{E}}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta, \vdash x \doteq_{\tau_1} y \longrightarrow s\ x =_{\tau_2} u\ y}\;\rightarrow\text{I}}{\text{(}x, y\text{ fresh)}\quad \mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta, \vdash \forall x\ y.\, x \doteq_{\tau_1} y \longrightarrow s\ x =_{\tau_2} u\ y}\;\forall\text{I}}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta, \vdash s =_{\tau_1 \Rightarrow \tau_2} u}\;\text{defn}\doteq
$$

Case $(\Pi(\alpha :_{\mathsf{R}} \mathsf{S}).\ \tau :_{\mathsf{R}} \mathsf{L})$:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma, s), \Gamma \vdash s \doteq_{\Pi\alpha.\ \tau} t}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma, s), \Gamma \vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow s\ [:\alpha:] \doteq_\tau t\ [:\alpha:]}\;\text{defn}\doteq}{[\alpha/\alpha]\quad \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma, s), \Gamma \vdash \mathcal{E}(\alpha) \longrightarrow s\ [:\alpha:] \doteq_\tau t\ [:\alpha:]}\;\forall_{\mathsf{ty}}\text{E + beta}}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma, s), \Gamma, \mathcal{E}(\alpha) \vdash s\ [:\alpha:] \doteq_\tau t\ [:\alpha:]}\;\rightarrow\text{E + assm}}{\mathcal{E}(t\ [:\alpha:]), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\tau), \mathcal{E}(\Gamma, s\ [:\alpha:]), \Gamma, \mathcal{E}(\alpha) \vdash s\ [:\alpha:] \doteq_\tau t\ [:\alpha:]}\;\text{defn}\mathcal{E}\text{ + weaken}^*}{\mathcal{E}(t\ [:\alpha:]), \mathbf{U}\mathcal{R}_{t\ [:\alpha:]}, \mathcal{E}(\tau), \mathcal{E}(\Gamma, s\ [:\alpha:]), \Gamma, \mathcal{E}(\alpha) \vdash s\ [:\alpha:] \doteq_\tau t\ [:\alpha:]}\;\text{defn}\mathcal{R}
\quad
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Delta, u), \Delta \vdash t \doteq_{\Pi\alpha.\ \tau} u}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Delta, u), \Delta \vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow t\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}\;\text{defn}\doteq}{[\alpha/\alpha]\quad \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Delta, u), \Delta \vdash \mathcal{E}(\alpha) \longrightarrow t\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}\;\forall_{\mathsf{ty}}\text{E + beta}}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Delta, u), \Delta, \mathcal{E}(\alpha) \vdash t\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}\;\rightarrow\text{E + assm}}{\mathcal{E}(t\ [:\alpha:]), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\tau), \mathcal{E}(\Delta, u\ [:\alpha:]), \Delta, \mathcal{E}(\alpha) \vdash t\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}\;\text{defn}\mathcal{E}\text{ + weaken}^*}{\mathcal{E}(t\ [:\alpha:]), \mathbf{U}\mathcal{R}_{t\ [:\alpha:]}, \mathcal{E}(\tau), \mathcal{E}(\Delta, u\ [:\alpha:]), \Delta, \mathcal{E}(\alpha) \vdash t\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}\;\text{defn}\mathcal{R}}{\mathcal{E}(\tau), \mathcal{E}(\Gamma, \Delta, s\ [:\alpha:], u\ [:\alpha:]), \Gamma, \Delta, \mathcal{E}(\alpha), \mathcal{E}(\alpha) \vdash s\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}\;\text{I.H.}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta, \mathcal{E}(\alpha) \vdash s\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}\;\text{contract}
$$

$$
\dfrac{\dfrac{\dfrac{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta, \mathcal{E}(\alpha) \vdash s\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash \mathcal{E}(\alpha) \longrightarrow s\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}\;\rightarrow\text{I}}{\text{(fresh }\alpha\text{)}\quad \mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow s\ [:\alpha:] \doteq_\tau u\ [:\alpha:]}\;\forall_{\mathsf{ty}}\text{I}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma, \Delta, s, u), \Gamma, \Delta \vdash s \doteq_{\Pi\alpha.\ \tau} u}\;\text{defn}\doteq
$$

Case α:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[s/x, t/y, u/z]\quad \dfrac{\dfrac{}{\mathcal{E}(\alpha) \vdash \forall x\ y\ z.\, x \doteq_\alpha y \longrightarrow y \doteq_\alpha z \longrightarrow x \doteq_\alpha z}\;\text{defn}\mathcal{E}\text{ + }\wedge\text{E}^+}{\mathcal{E}(\alpha) \vdash s \doteq_\alpha t \longrightarrow t \doteq_\alpha u \longrightarrow s \doteq_\alpha u}\;(\forall\text{E + beta})^+ \quad \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\alpha), \mathcal{E}(\Gamma, s), \Gamma \vdash s \doteq_\alpha t \quad \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\alpha), \mathcal{E}(\Delta, u), \Delta \vdash t \doteq_\alpha u}{\mathcal{E}(t), \mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathbf{U}\mathcal{R}_t, \mathcal{E}(\alpha), \mathcal{E}(\alpha), \mathcal{E}(\Gamma, s), \mathcal{E}(\Delta, u), \Gamma, \Delta \vdash s \doteq_\alpha u}\;\rightarrow\text{E}^+}{\mathcal{E}(t), \mathbf{U}\mathcal{R}_t, \mathcal{E}(\alpha), \mathcal{E}(\Gamma, \Delta, u, s), \Gamma, \Delta \vdash s \doteq_\alpha u}\;\text{contract}}{\mathcal{E}(t), \mathcal{E}(\mathbf{U}\mathcal{R}_t), \mathcal{E}(\alpha), \mathcal{E}(\Gamma, \Delta, u, s), \Gamma, \Delta \vdash s \doteq_\alpha u}\;\text{drop-unused}\mathcal{R}}{\mathbf{U}\mathcal{E}_t, \mathcal{E}(\alpha), \mathcal{E}(\Gamma, \Delta, u, s), \Gamma, \Delta \vdash s \doteq_\alpha u}\;\text{contract}}{\mathcal{E}(\alpha), \mathcal{E}(\Gamma, \Delta, u, s), \Gamma, \Delta \vdash s \doteq_\alpha u}\;\text{drop-unused}\mathcal{E}
$$

□

C.1.7 mk-comb

The rule mk-comb is

$$
\dfrac{\Gamma \vdash s = t \qquad \Delta \vdash u = v}{\Gamma, \Delta \vdash s\ u = t\ v}\ \text{mk-comb}
$$

translated, this rule is

$$
\dfrac{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_{\tau_1 \Rightarrow \tau_2} t^\bullet \qquad \mathcal{E}(\tau_1), \mathcal{E}(\Delta^\bullet, u^\bullet, v^\bullet), \mathcal{R}(\Delta^\bullet, u^\bullet, v^\bullet), \Delta^\bullet, \vdash u^\bullet \doteq_{\tau_1} v^\bullet}{\mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, (s\ u)^\bullet, (t\ v)^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, (s\ u)^\bullet, (t\ v)^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash (s\ u)^\bullet \doteq_{\tau_2} (t\ v)^\bullet}\ \text{mk-comb}\bullet
$$

which follows from *mk-comb*$\doteq$ (Subsection B.2.10).

$$\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_{\tau_1 \Rightarrow \tau_2} t^\bullet \quad \mathcal{E}(\tau_1), \mathcal{E}(\Delta^\bullet, u^\bullet, v^\bullet), \mathcal{R}(\Delta^\bullet, u^\bullet, v^\bullet), \Delta^\bullet, \vdash u^\bullet \doteq_{\tau_1} v^\bullet}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{E}(\tau_1), \mathcal{E}(\Delta^\bullet, u^\bullet, v^\bullet), \mathcal{R}(\Delta^\bullet, u^\bullet, v^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash s^\bullet\ u^\bullet \doteq_{\tau_2} t^\bullet\ v^\bullet}\ \text{defn}\mathcal{E}}{\mathcal{E}(\tau_1), \mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{E}(\tau_1), \mathcal{E}(\Delta^\bullet, u^\bullet, v^\bullet), \mathcal{R}(\Delta^\bullet, u^\bullet, v^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash s^\bullet\ u^\bullet \doteq_{\tau_2} t^\bullet\ v^\bullet}\ \text{drop-unused}\mathcal{E}}{\mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{E}(\Delta^\bullet, u^\bullet, v^\bullet), \mathcal{R}(\Delta^\bullet, u^\bullet, v^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash s^\bullet\ u^\bullet \doteq_{\tau_2} t^\bullet\ v^\bullet}\ \text{defn}\bullet + \text{defn}\mathcal{E} + \text{defn}\mathcal{R}}{\mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, (s\ u)^\bullet, (t\ v)^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, (s\ u)^\bullet, (t\ v)^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash (s\ u)^\bullet \doteq_{\tau_2} (t\ v)^\bullet}\ \text{mk-comb}\doteq$$

C.1.8 beta

The rule beta is

$$\dfrac{}{\vdash (\lambda x.\ s)\ t =_\tau s[t/x]}\ \text{beta}$$

translated, this rule is

$$\dfrac{}{\mathcal{E}(\tau), \mathcal{E}(((\lambda x.\ s)\ t)^\bullet, (s[t/x])^\bullet), \mathcal{R}(((\lambda x.\ s)\ t)^\bullet, (s[t/x])^\bullet) \vdash ((\lambda x.\ s)\ t)^\bullet \doteq_\tau (s[t/x])^\bullet}$$

The proof is

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash (\lambda x.\ s^\bullet)\ t^\bullet \doteq_\tau s^\bullet[t^\bullet/x])}\ \text{beta} \quad \dfrac{}{\mathcal{E}(\tau), \mathcal{E}((\lambda x.\ s^\bullet)\ t^\bullet), \mathcal{R}((\lambda x.\ s^\bullet)\ t^\bullet) \vdash (\lambda x.\ s^\bullet)\ t^\bullet \doteq_\tau (\lambda x.\ s^\bullet)\ t^\bullet}\ \text{refl}\bullet}{\mathcal{E}(\tau), \mathcal{E}((\lambda x.\ s^\bullet)\ t), \mathcal{R}((\lambda x.\ s^\bullet)\ t) \vdash (\lambda x.\ s^\bullet)\ t^\bullet \doteq_\tau s^\bullet[t^\bullet/x])}\ \text{rewrite}}{\mathcal{E}(\tau), \mathcal{E}(((\lambda x.\ s)\ t)^\bullet), \mathcal{R}(((\lambda x.\ s)\ t)^\bullet) \vdash ((\lambda x.\ s)\ t)^\bullet \doteq_\tau (s[t/x])^\bullet}\ \text{defn}\bullet + \text{subst}\bullet}{\mathcal{E}(\tau), \mathcal{E}(((\lambda x.\ s)\ t)^\bullet), (s[t/x])^\bullet), \mathcal{R}(((\lambda x.\ s)\ t)^\bullet, (s[t/x])^\bullet) \vdash ((\lambda x.\ s)\ t)^\bullet \doteq_\tau (s[t/x])^\bullet}\ \text{weaken}$$

C.1.9 abs

Recall the proof rule abs is

$$(z \text{ not in } \Gamma)\ \dfrac{\Gamma \vdash s = t}{\Gamma \vdash (\lambda z.\ s) = (\lambda z.\ t)}\ \text{abs}$$

translated, this rule is

$$(z \text{ not in } \Gamma^\bullet)\ \dfrac{\mathcal{E}(\tau_2), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_{\tau_2} t^\bullet}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}(\Gamma^\bullet, (\lambda z.\ s)^\bullet, (\lambda z.\ t)^\bullet), \mathcal{R}(\Gamma^\bullet, (\lambda z.\ s)^\bullet, (\lambda z.\ t)^\bullet), \Gamma^\bullet \vdash (\lambda z.\ s)^\bullet \doteq_{\tau_1 \Rightarrow \tau_2} (\lambda z.\ t)^\bullet}\ \text{abs}\bullet$$

The proof is as follows

C.1.10 eta

The rule eta is

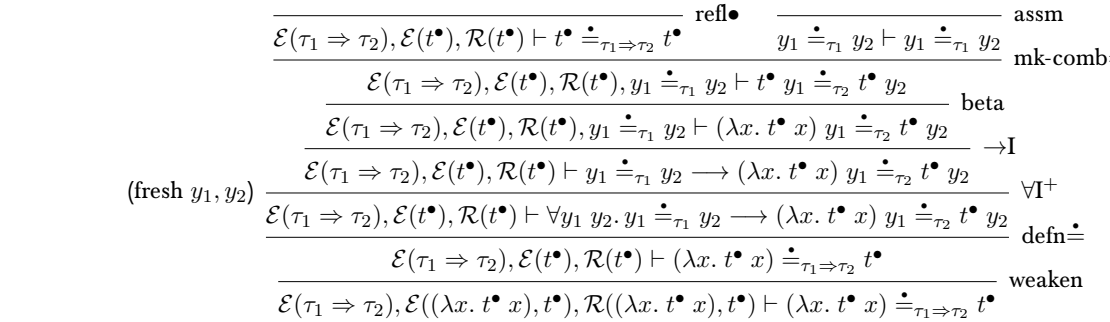
$$(x \text{ not in } t)\ \frac{}{\vdash (\lambda x.\ t\ x) =_{\tau_1 \Rightarrow \tau_2} t}\ \text{eta}$$

we show that the translated proof rule

$$(x \text{ not in } t^\bullet)\ \frac{}{\mathcal{E}(\tau_1 \Rightarrow \tau_2), \mathcal{E}((\lambda x.\ t^\bullet\ x), t^\bullet), \mathcal{R}((\lambda x.\ t^\bullet\ x), t^\bullet) \vdash (\lambda x.\ t^\bullet\ x) \doteq_{\tau_1 \Rightarrow \tau_2} t^\bullet}\ \text{eta}\bullet$$

is derivable in PHOL.

The proof of eta translation is as follows



C.1.11 tyapp

The rule tyapp is

$$\frac{\Gamma \vdash s =_{\Lambda\alpha.\ \tau} t}{\Gamma, \Delta \vdash s\ [:\alpha:] =_\tau t\ [:\alpha:]}\ \text{tyapp}$$

translated, this rule is

$$\frac{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet, \vdash s^\bullet \doteq_{\Pi\alpha.\ \tau} t^\bullet}{\mathcal{E}(\tau), \mathcal{E}(\Gamma^\bullet, (s\ [:\alpha:])^\bullet, (t\ [:\alpha:])^\bullet), \mathcal{R}(\Gamma^\bullet, (s\ [:\alpha:])^\bullet, (t\ [:\alpha:])^\bullet), \Gamma^\bullet \vdash (s\ [:\alpha:])^\bullet \doteq_\tau (t\ [:\alpha:])^\bullet}\ \text{tyapp}\bullet$$

To prove this, we use the rule *tyapp*$\doteq$ (B.2.11)

$$\frac{\Gamma \vdash s^\bullet \doteq_{\Lambda\alpha.\ \tau} t^\bullet}{\mathcal{E}(\alpha), \Gamma \vdash s^\bullet\ [:\alpha:] \doteq_\tau t^\bullet\ [:\alpha:]}\ \text{tyapp}\doteq$$

The proof is as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet, \vdash s^\bullet \doteq_{\Pi\alpha.\ \tau} t^\bullet}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\alpha), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet\ [:\alpha:] \doteq_\tau t^\bullet\ [:\alpha:]}\ \text{tyapp}\doteq}{\mathcal{E}(\tau), \mathcal{E}(\alpha), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet\ [:\alpha:] \doteq_\tau t^\bullet\ [:\alpha:]}\ \text{weaken + defn}\mathcal{E}}{\mathcal{E}(\tau), \mathcal{E}(\alpha), \mathcal{E}(\alpha), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet\ [:\alpha:] \doteq_\tau t^\bullet\ [:\alpha:]}\ \text{weaken}}{\mathcal{E}(\tau), \mathcal{E}(\Gamma^\bullet, s^\bullet\ [:\alpha:], t^\bullet\ [:\alpha:]), \mathcal{R}(\Gamma^\bullet, s^\bullet\ [:\alpha:], t^\bullet\ [:\alpha:]), \Gamma^\bullet \vdash s^\bullet\ [:\alpha:] \doteq_\tau t^\bullet\ [:\alpha:]}\ \text{defn}\mathcal{R} + \text{defn}\mathcal{E}}{\mathcal{E}(\tau), \mathcal{E}(\Gamma^\bullet, (s\ [:\alpha:])^\bullet, (t\ [:\alpha:])^\bullet), \mathcal{R}(\Gamma^\bullet, (s\ [:\alpha:])^\bullet, (t\ [:\alpha:])^\bullet), \Gamma^\bullet \vdash (s\ [:\alpha:])^\bullet \doteq_\tau (t\ [:\alpha:])^\bullet}\ \text{defn}\bullet$$

Note that $\mathcal{E}(\Pi\alpha.\ \tau)$ produces $\mathcal{E}(\tau)$ in the following manner. By cases: if $\mathrm{FV}_{\mathrm{ty}}(\tau)$ contains α, then we may use *weaken* to generate these assumptions; if it does not, then $\mathcal{E}(\Pi\alpha.\ \tau)$ is equal to $\mathcal{E}(\tau)$.

C.1.12 tybeta

The rule tybeta is

$$\frac{}{\vdash (\Lambda\alpha.\ s)\ [:\sigma:] =_\tau s[\sigma/\alpha]}\ \text{tybeta}$$

Translated, this rule is

$$\frac{}{\mathcal{E}(\tau), \mathcal{E}(((\Lambda\alpha.\ s)\ [:\sigma:]), (s[\sigma/\alpha])), \mathcal{R}(((\Lambda\alpha.\ s)\ [:\sigma:]), (s[\sigma/\alpha])) \vdash ((\Lambda\alpha.\ s)\ [:\sigma:])^\bullet \doteq_\tau (s[\sigma/\alpha])^\bullet}$$

Proof:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash (\Lambda\alpha.\ s^\bullet)\ [:\sigma:] =_\tau (s^\bullet[\sigma/\alpha])}\ \text{tybeta} \quad \dfrac{}{\mathcal{E}(\tau), \mathcal{E}((\Lambda\alpha.\ s)\ [:\sigma:]), \mathcal{R}((\Lambda\alpha.\ s)\ [:\sigma:]) \vdash (\Lambda\alpha.\ s^\bullet)\ [:\sigma:] \doteq_\tau (\Lambda\alpha.\ s^\bullet)\ [:\sigma:]}\ \text{refl}\bullet}{\mathcal{E}(\tau), \mathcal{E}((\Lambda\alpha.\ s)\ [:\sigma:]), \mathcal{R}((\Lambda\alpha.\ s)\ [:\sigma:]) \vdash (\Lambda\alpha.\ s^\bullet)\ [:\sigma:] \doteq_\tau (s^\bullet[\sigma/\alpha])}\ \text{rewrite}}{\mathcal{E}(\tau), \mathcal{E}((\Lambda\alpha.\ s)\ [:\sigma:]), \mathcal{R}((\Lambda\alpha.\ s)\ [:\sigma:]) \vdash ((\Lambda\alpha.\ s)\ [:\sigma:])^\bullet \doteq_\tau (s[\sigma/\alpha])^\bullet}\ \text{defn}\bullet + \text{tysubst}\bullet}{\mathcal{E}(\tau), \mathcal{E}(((\Lambda\alpha.\ s)\ [:\sigma:]), (s[\sigma/\alpha])), \mathcal{R}(((\Lambda\alpha.\ s)\ [:\sigma:]), (s[\sigma/\alpha])) \vdash ((\Lambda\alpha.\ s)\ [:\sigma:])^\bullet \doteq_\tau (s[\sigma/\alpha])^\bullet}\ \text{weaken}$$

C.1.13 tyabs

The rule *tyabs* is

$$(\alpha \text{ not in } \Gamma)\ \frac{\Gamma \vdash s =_\tau t}{\Gamma \vdash (\Lambda\alpha.\ s) =_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)}\ \text{tyabs}$$

translated, this rule is

$$(\alpha \text{ not in } \Gamma^\bullet)\ \frac{\mathcal{E}(\tau), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_\tau t^\bullet}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s)^\bullet, (\Lambda\alpha.\ t)^\bullet), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s)^\bullet, (\Lambda\alpha.\ t)^\bullet), \Gamma^\bullet \vdash (\Lambda\alpha.\ s)^\bullet \doteq_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)^\bullet}\ \text{tyabs}\bullet$$

The proof of this rule is as follows

$$(\alpha \text{ fresh})\ \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}(\tau), \mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_\tau t^\bullet}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\alpha), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_\tau t^\bullet}\ \text{weaken}^* + \text{contract}^*}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\alpha), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \Gamma^\bullet \vdash s^\bullet \doteq_\tau t^\bullet}\ \text{defn}\mathcal{R}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\alpha), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \Gamma^\bullet \vdash (\Lambda\alpha.\ s^\bullet)\ [:\alpha:] \doteq_\tau (\Lambda\alpha.\ t^\bullet)\ [:\alpha:]}\ \text{tybeta}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \Gamma^\bullet \vdash \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ s^\bullet)\ [:\alpha:] \doteq_\tau (\Lambda\alpha.\ t^\bullet)\ [:\alpha:]}\ \rightarrow\text{I}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \Gamma^\bullet \vdash \forall_{\text{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ s^\bullet)\ [:\alpha:] \doteq_\tau (\Lambda\alpha.\ t^\bullet)\ [:\alpha:]}\ \forall_{\text{ty}}\text{I}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet)), \Gamma^\bullet \vdash (\Lambda\alpha.\ s^\bullet) \doteq_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t^\bullet)}\ \text{defn}\doteq}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s)^\bullet, (\Lambda\alpha.\ t)^\bullet), \mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s)^\bullet, (\Lambda\alpha.\ t)^\bullet), \Gamma^\bullet \vdash (\Lambda\alpha.\ s)^\bullet \doteq_{\Pi\alpha.\ \tau} (\Lambda\alpha.\ t)^\bullet}\ \text{defn}\bullet$$

Note that, as $\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet))$ and $\mathcal{R}(\Gamma^\bullet, (\Lambda\alpha.\ s^\bullet), (\Lambda\alpha.\ t^\bullet))$ do not contain α, the condition 'α fresh' reduces to 'α not in $\Gamma^\bullet$'.

C.1.14 tyeta

The rule tyeta is

$$(\alpha \text{ not in } f)\ \frac{}{\vdash (\Lambda\alpha.\ f\ [:\alpha:]) \doteq_{\Pi\alpha.\ \tau} f}\ \text{tyeta}$$

Translated, this rule is

$$(\alpha \text{ not in } f^\bullet)\ \frac{}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]), f^\bullet), \mathcal{R}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]), f^\bullet) \vdash (\Lambda\alpha.\ f^\bullet\ [:\alpha:]) \doteq_{\Pi\alpha.\ \tau} f^\bullet}\ \text{tyeta}$$

Proof:

$$\dfrac{(\alpha \text{ fresh})\ \dfrac{(\alpha \text{ not in } f^\bullet)\ \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{E}(\tau), \mathcal{E}((f^\bullet\ [:\alpha:])^\bullet), \mathcal{R}((f\ [:\alpha:])^\bullet) \vdash (f\ [:\alpha:])^\bullet \doteq_\tau (f\ [:\alpha:])^\bullet}\ \text{refl}\bullet}{\mathcal{E}(\tau), \mathcal{E}(f^\bullet\ [:\alpha:]), \mathcal{R}(f^\bullet\ [:\alpha:]) \vdash f^\bullet\ [:\alpha:] \doteq_\tau f^\bullet\ [:\alpha:]}\ \text{defn}\bullet}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(f^\bullet), \mathcal{R}(f^\bullet), \mathcal{E}(\alpha) \vdash f^\bullet\ [:\alpha:] \doteq_\tau f^\bullet\ [:\alpha:]}\ \text{defn}\mathcal{E} + \text{defn}\mathcal{R}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(f^\bullet), \mathcal{R}(f^\bullet), \mathcal{E}(\alpha) \vdash (\Lambda\alpha.\ f^\bullet\ [:\alpha:])\ [:\alpha:] \doteq_\tau f^\bullet\ [:\alpha:]}\ \text{type-beta}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(f^\bullet), \mathcal{R}(f^\bullet) \vdash \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ f^\bullet\ [:\alpha:])\ [:\alpha:] \doteq_\tau f^\bullet\ [:\alpha:]}\ \to\text{I}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(f^\bullet), \mathcal{R}(f^\bullet) \vdash \forall_{\text{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow (\Lambda\alpha.\ f^\bullet\ [:\alpha:])\ [:\alpha:] \doteq_\tau f^\bullet\ [:\alpha:]}\ \forall_{\text{ty}}\text{I}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(f^\bullet), \mathcal{R}(f^\bullet) \vdash (\Lambda\alpha.\ f^\bullet\ [:\alpha:]) \doteq_{\Pi\alpha.\ \tau} f^\bullet}\ \text{defn}\doteq$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}(f^\bullet, f^\bullet), \mathcal{R}(f^\bullet, f^\bullet) \vdash (\Lambda\alpha.\ f^\bullet\ [:\alpha:]) \doteq_{\Pi\alpha.\ \tau} f^\bullet}\ \text{weaken}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]), f^\bullet), \mathcal{R}(f^\bullet) \vdash (\Lambda\alpha.\ f^\bullet\ [:\alpha:]) \doteq_{\Pi\alpha.\ \tau} f^\bullet}\ \text{defn}\mathcal{E}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]), f^\bullet), \mathcal{R}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]), f^\bullet) \vdash (\Lambda\alpha.\ f^\bullet\ [:\alpha:]) \doteq_{\Pi\alpha.\ \tau} f^\bullet}\ \text{defn}\mathcal{R}}{\mathcal{E}(\Pi\alpha.\ \tau), \mathcal{E}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]), f^\bullet), \mathcal{R}((\Lambda\alpha.\ f^\bullet\ [:\alpha:]), f^\bullet) \vdash (\Lambda\alpha.\ f\ [:\alpha:])^\bullet \doteq_{\Pi\alpha.\ \tau} f^\bullet}\ \text{defn}\bullet$$

C.1.15 eq-mp

The rule *eq-mp* is

$$\frac{\Gamma \vdash s =_o t \qquad \Delta \vdash s}{\Gamma, \Delta \vdash t}\ \text{eq-mp}$$

Translated, this rule is

$$\frac{\mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_o t^\bullet \qquad \mathcal{E}(\Delta^\bullet, s^\bullet), \mathcal{R}(\Delta^\bullet, s^\bullet), \Delta^\bullet \vdash s^\bullet}{\mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{eq-mp}\bullet$$

To begin, define the following abbreviations

$$\begin{aligned}
\mathbf{P}\mathcal{E}_{s^\bullet} &\triangleq \mathcal{E}(\mathrm{FV}_{\mathrm{ty}}(s^\bullet) \cap \mathrm{FV}_{\mathrm{ty}}(\Gamma^\bullet, \Delta^\bullet, t^\bullet)) \\
\mathbf{U}\mathcal{E}_{s^\bullet} &\triangleq \mathcal{E}(\mathrm{FV}_{\mathrm{ty}}(s^\bullet) - \mathrm{FV}_{\mathrm{ty}}(\Gamma^\bullet, \Delta^\bullet, t^\bullet)) \\
\mathbf{P}\mathcal{R}_{s^\bullet} &\triangleq \mathcal{R}(\mathrm{FV}(s^\bullet) \cap \mathrm{FV}(\Gamma^\bullet, \Delta^\bullet, t^\bullet)) \\
\mathbf{U}\mathcal{R}_{s^\bullet} &\triangleq \mathcal{R}(\mathrm{FV}(s^\bullet) - \mathrm{FV}(\Gamma^\bullet, \Delta^\bullet, t^\bullet))
\end{aligned}$$

$$\begin{aligned}
\mathbf{P}\mathcal{E}_{\mathbf{U}\mathcal{R}_{s^\bullet}} &\triangleq \mathcal{E}(\mathrm{FV}_{\mathrm{ty}}(\mathbf{U}\mathcal{R}_{s^\bullet}) \cap \mathrm{FV}_{\mathrm{ty}}(\Gamma^\bullet, \Delta^\bullet, t^\bullet)) \\
\mathbf{U}\mathcal{E}_{\mathbf{U}\mathcal{R}_{s^\bullet}} &\triangleq \mathcal{E}(\mathrm{FV}_{\mathrm{ty}}(\mathbf{U}\mathcal{R}_{s^\bullet}) - \mathrm{FV}_{\mathrm{ty}}(\Gamma^\bullet, \Delta^\bullet, t^\bullet))
\end{aligned}$$

From these, it follows by he definition of $\mathcal{E}$ and $\mathcal{R}$ and set arithmetic that

$$\begin{aligned}
\mathcal{E}(s^\bullet) &\equiv \mathbf{P}\mathcal{E}_{s^\bullet} \cup \mathbf{U}\mathcal{E}_{s^\bullet} \\
\mathcal{R}(s^\bullet) &\equiv \mathbf{P}\mathcal{R}_{s^\bullet} \cup \mathbf{U}\mathcal{R}_{s^\bullet}
\end{aligned}$$

$$\begin{aligned}
\mathbf{P}\mathcal{E}_{\mathbf{U}\mathcal{R}_{s^\bullet}} &\subseteq \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet) \\
\mathbf{U}\mathcal{E}_{\mathbf{U}\mathcal{R}_{s^\bullet}} &\subseteq \mathbf{U}\mathcal{E}_{s^\bullet}
\end{aligned}$$

The proof is as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet \doteq_o t^\bullet}{\mathcal{E}(\Gamma^\bullet, s^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, s^\bullet, t^\bullet), \Gamma^\bullet \vdash s^\bullet =_o t^\bullet}\ \text{defn}\doteq \quad \mathcal{E}(\Delta^\bullet), \mathcal{R}(\Delta^\bullet), \mathcal{E}(s^\bullet), \mathcal{R}(s^\bullet), \Delta^\bullet \vdash s^\bullet}{\mathcal{E}(\Delta^\bullet), \mathcal{R}(\Delta^\bullet), \mathcal{E}(s^\bullet), \mathcal{R}(s^\bullet), \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{eq-mp}}{\mathcal{E}(s^\bullet), \mathcal{R}(s^\bullet), \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{contract}}{\mathbf{U}\mathcal{R}_{s^\bullet}, \mathbf{U}\mathcal{E}_{s^\bullet}, \mathbf{P}\mathcal{R}_{s^\bullet}, \mathbf{P}\mathcal{E}_{s^\bullet}, \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{defn}\mathcal{E} + \text{defn}\mathcal{R}}{\mathbf{U}\mathcal{E}_{s^\bullet}, \mathbf{U}\mathcal{R}_{s^\bullet}, \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{contract}}{\mathbf{U}\mathcal{E}_{s^\bullet}, \mathcal{E}(\mathbf{U}\mathcal{R}_{s^\bullet}), \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{defn}\mathcal{E}}{\mathbf{U}\mathcal{E}_{s^\bullet}, \mathbf{U}\mathcal{E}_{\mathbf{U}\mathcal{R}_{s^\bullet}}, \mathbf{P}\mathcal{E}_{\mathbf{U}\mathcal{R}_{s^\bullet}} \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{drop-unused}\mathcal{R}}{\mathbf{U}\mathcal{E}_{s^\bullet}, \mathbf{U}\mathcal{E}_{\mathbf{U}\mathcal{R}_{s^\bullet}}, \mathbf{P}\mathcal{E}_{\mathbf{U}\mathcal{R}_{s^\bullet}} \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{contract}}{\mathbf{U}\mathcal{E}_{s^\bullet}, \mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}\ \text{drop-unused}\mathcal{E}}{\mathcal{E}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, t^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash t^\bullet}$$

C.1.16 deduct-antisym

The rule deduct-antisym is

$$\dfrac{\Gamma, q \vdash p \qquad \Delta, p \vdash q}{\Gamma, \Delta \vdash p =_o q}\ \text{deduct-antisym}$$

translated, this rule is

$$\dfrac{\mathcal{E}(\Gamma^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Gamma^\bullet, p^\bullet, q^\bullet), \Gamma^\bullet, q^\bullet \vdash p^\bullet \qquad \mathcal{E}(\Delta^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Delta^\bullet, p^\bullet, q^\bullet), \Delta^\bullet, p^\bullet \vdash q^\bullet}{\mathcal{E}(\Gamma^\bullet, \Delta^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, p^\bullet, q^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash p^\bullet =_o q^\bullet}\ \text{deduct-antisym}\bullet$$

The proof of this rule is as follows

$$\dfrac{\dfrac{\dfrac{\mathcal{E}(\Gamma^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Gamma^\bullet, p^\bullet, q^\bullet), \Gamma^\bullet, q^\bullet \vdash p^\bullet \qquad \mathcal{E}(\Delta^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Delta^\bullet, p^\bullet, q^\bullet), \Delta^\bullet, p^\bullet \vdash q^\bullet}{\mathcal{E}(\Gamma^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Gamma^\bullet, p^\bullet, q^\bullet), \mathcal{E}(\Delta^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Delta^\bullet, p^\bullet, q^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash p^\bullet =_o q^\bullet}\ \text{eq-mp}}{\mathcal{E}(\Gamma^\bullet, \Delta^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, p^\bullet, q^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash p^\bullet =_o q^\bullet}\ \text{contract}}{\mathcal{E}(\Gamma^\bullet, \Delta^\bullet, p^\bullet, q^\bullet), \mathcal{R}(\Gamma^\bullet, \Delta^\bullet, p^\bullet, q^\bullet), \Gamma^\bullet, \Delta^\bullet \vdash p^\bullet \doteq_o q^\bullet}\ \text{defn}\doteq$$

C.1.17 select-ax

The rule 'select-ax' is

$$\dfrac{}{\vdash p\ x \longrightarrow p\ (\varepsilon x.\, p\ x)}\ \text{select-ax}$$

translated, this rule is

$$\dfrac{}{\mathcal{E}((p\ x \longrightarrow p\ (\varepsilon x.\, p\ x))^\bullet), \mathcal{R}((p\ x \longrightarrow p\ (\varepsilon x.\, p\ x))^\bullet) \vdash (p\ x \longrightarrow p\ (\varepsilon x.\, p\ x))^\bullet}\ \text{select-ax}\bullet$$

Note that this proof relies on the fact that $\varepsilon^\bullet \equiv \varepsilon$ and $(s \longrightarrow t)^\bullet = s^\bullet \longrightarrow t^\bullet$.

Proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash p^\bullet\ x \longrightarrow p^\bullet\ (\varepsilon x.\, p^\bullet\ x)}\ \text{select-ax}}{\vdash p^\bullet\ x \longrightarrow p^\bullet\ (\varepsilon x.\, p\ x)^\bullet}\ \varepsilon\bullet + \text{defn}\bullet}{\vdash (p\ x \longrightarrow p\ (\varepsilon x.\, p\ x))^\bullet}\ \longrightarrow\bullet + \text{defn}\bullet}{\mathcal{E}((p\ x \longrightarrow p\ (\varepsilon x.\, p\ x))^\bullet), \mathcal{R}((p\ x \longrightarrow p\ (\varepsilon x.\, p\ x))^\bullet) \vdash (p\ x \longrightarrow p\ (\varepsilon x.\, p\ x))^\bullet}\ \text{weaken}$$

□

C.1.18 infinity-ax

The rule 'infinity-ax' is

$$\dfrac{}{\vdash \exists(f : \iota \Rightarrow \iota).\, \mathrm{inj}\ f \wedge \neg\, \mathrm{onto}\ f}\ \text{infinity-ax}$$

translated, this rule is

$$\dfrac{}{\vdash (\exists(f : \iota \Rightarrow \iota).\, \mathrm{inj}\ f \wedge \neg\, \mathrm{onto}\ f)^\bullet}\ \text{infinity-ax}\bullet$$

Note that the theorem has no free variables or free type-variables, as such, its $\mathcal{E}$ and $\mathcal{R}$ are both empty.

This lemma depends on the translation of onto, and inj, and these in turn rely on the translation of $[\forall]$ and $[\exists]$. A crucial point to note is that becase $\vdash (f \doteq_{\iota \Rightarrow \iota} f)$ and $\top \longrightarrow$

$p = p$, that the additional complicating assumptions can be removed from the rules $\forall\bullet$ and $\exists\bullet$. The proof of infinity axiom is then by simply unfolding the definition $\bullet$.

Here is the proof that $f \stackrel{\bullet}{=}_{\iota\Rightarrow\iota} f$ is true:

$$(x, y \text{ fresh})\ \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash f\ x =_\iota f\ x}\ \text{refl}}{x =_\iota y \vdash f\ x =_\iota f\ y}\ \text{rewrite + assm}}{\vdash x =_\iota y \longrightarrow f\ x =_\iota f\ y}\ \to\text{I}}{\vdash \forall x\ y.\ x =_\iota y \longrightarrow f\ x =_\iota f\ y}\ \forall\text{I}}{\vdash f \stackrel{\bullet}{=}_{\iota\Rightarrow\iota} f}\ \text{defn}\stackrel{\bullet}{=}$$

C.2 Not Trivial

The proof that the translation is non-trivial is in three steps.

C.2.1 Step 1: Mapping Backwards

We first define the mapping $(-)^\circ$ from PHOL terms to $\text{PHOL}_{\xi\eta}$ terms.

$$\begin{array}{lcl}
x^\circ & \triangleq & x \\
(\lambda x.\ s)^\circ & \triangleq & \lambda x.\ s^\circ \\
(s\ t)^\circ & \triangleq & s^\circ\ t^\circ \\
(\Lambda\alpha.\ s)^\circ & \triangleq & \Lambda\alpha.\ s^\circ \\
(s\ [\!:\tau:\!])^\circ & \triangleq & s^\circ\ [\!:\tau:\!] \\
([=]_\tau)^\circ & \triangleq & [=]_\tau \\
([\stackrel{\bullet}{=}]_\alpha)^\circ & \triangleq & [=]_\alpha \\
([\forall]_\tau)^\circ & \triangleq & [\forall]_\tau \\
([\forall_{\mathsf{ty}}])^\circ & \triangleq & [\forall_{\mathsf{ty}}]
\end{array}$$

we will use this to show the theorem

$$\dfrac{}{\vdash_{\text{PHOL}_{\xi\eta}} s^{\bullet\circ} = s}\ \text{translate-reverse-eq-id}$$

Proof. By induction on s.

Case x:

$$\dfrac{\dfrac{\dfrac{}{\vdash x = x}\ \text{refl}}{\vdash x^\circ = x}\ \text{defn}\circ}{\vdash x^{\bullet\circ} = x}\ \text{defn}\bullet$$

Case $\lambda x.\ s$:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash s^{\bullet\circ} = s}\ \text{I.H.}}{\vdash (\lambda x.\ s^{\bullet\circ}) = (\lambda x.\ s)}\ \text{abs}}{\vdash (\lambda x.\ s^{\bullet})^{\circ} = (\lambda x.\ s)}\ \text{defn}\circ}{\vdash (\lambda x.\ s)^{\bullet\circ} = (\lambda x.\ s)}\ \text{defn}\bullet$$

Case $s\ t$:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash s^{\bullet\circ} = s}\ \text{I.H.} \quad \dfrac{}{\vdash t^{\bullet\circ} = t}\ \text{I.H.}}{\vdash s^{\bullet\circ}\ t^{\bullet\circ} = s\ t}\ \text{mk-comb}}{\vdash (s^{\bullet}\ t^{\bullet})^{\circ} = s\ t}\ \text{defn}\circ}{\vdash (s\ t)^{\bullet\circ} = s\ t}\ \text{defn}\bullet$$

Case $\Lambda\alpha.\ s$:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash s^{\bullet\circ} = s}\ \text{I.H.}}{\vdash (\Lambda\alpha.\ s^{\bullet\circ}) = (\Lambda\alpha.\ s)}\ \text{tyabs}}{\vdash (\Lambda\alpha.\ s^{\bullet})^{\circ} = (\Lambda\alpha.\ s)}\ \text{defn}\circ}{\vdash (\Lambda\alpha.\ s)^{\bullet\circ} = (\Lambda\alpha.\ s)}\ \text{defn}\bullet$$

Case $s\ [\!:\alpha:\!]$:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash s^{\bullet\circ} = s}\ \text{I.H.}}{\vdash s^{\bullet\circ}\ [\!:\alpha:\!] = s\ [\!:\alpha:\!]}\ \text{tyapp}}{\vdash (s^{\bullet}\ [\!:\alpha:\!])^{\circ} = s\ [\!:\alpha:\!]}\ \text{defn}\circ}{\vdash (s\ [\!:\alpha:\!])^{\bullet\circ} = s\ [\!:\alpha:\!]}\ \text{defn}\bullet$$

Case $[=]_\tau$:

By cases on τ.

Subcase $[=]_\chi ([=]_o / [=]_\iota)$:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash [=]_\chi = [=]_\chi}\ \text{refl}}{\vdash [=]_\chi^{\circ} = [=]_\chi}\ \text{defn}\circ}{\vdash [\dot{=}]_\chi^{\circ} = [=]_\chi}\ \text{defn}\dot{=}}{\vdash [=]_\chi^{\bullet\circ} = [=]_\chi}\ \text{defn}\bullet$$

Subcase $[=]_{\tau_1 \Rightarrow \tau_2}$:

$$[x/x, x/y]\ \dfrac{\dfrac{\dfrac{\dfrac{}{\forall x\ y.\, x =_{\tau_1} y \longrightarrow f\ x =_{\tau_2} g\ y \vdash \forall x\ y.\, x =_{\tau_1} y \longrightarrow f\ x =_{\tau_2} g\ y}\ \text{assm}}{\forall x\ y.\, x =_{\tau_1} y \longrightarrow f\ x =_{\tau_2} g\ y \vdash x =_{\tau_1} x \longrightarrow f\ x =_{\tau_2} g\ x}\ \forall\text{E} \qquad \dfrac{}{\vdash x = x}\ \text{refl}}{\forall x\ y.\, x =_{\tau_1} y \longrightarrow f\ x =_{\tau_2} g\ y \vdash f\ x = g\ x}\ \text{fun-ext}}{\forall x\ y.\, x =_{\tau_1} y \longrightarrow f\ x =_{\tau_2} g\ y \vdash f = g}$$

$$\vdots\ 1$$

$$\dfrac{\dfrac{\dfrac{}{f = g \vdash f = g}\ \text{assm} \qquad \dfrac{}{x =_{\tau_1} y \vdash x =_{\tau_1} y}\ \text{assm} \qquad \dfrac{}{\vdash g\ y =_{\tau_2} g\ y}\ \text{refl}}{f = g, x =_{\tau_1} y \vdash f\ x =_{\tau_2} g\ y}\ \text{rewrite}^+}{f = g \vdash x =_{\tau_1} y \longrightarrow f\ x =_{\tau_2} g\ y}\ \to\text{I}}{f = g \vdash \forall x\ y.\, x =_{\tau_1} y \longrightarrow f\ x =_{\tau_2} g\ y}\ \forall\text{I}\ (x, y \text{ fresh})$$

$$\vdots\ 2$$

$$\dfrac{\vdots\ 1 \qquad \vdots\ 2}{\forall x\ y.[=]_{\tau_1}\ x\ y \longrightarrow [=]_{\tau_2}\ (f\ x)\ (g\ y) \vdash f = g \qquad f = g \vdash \forall x\ y.[=]_{\tau_1}\ x\ y \longrightarrow [=]_{\tau_2}\ (f\ x)\ (g\ y)}\ \text{deduct-antisym}$$

$$\vdash \big(\forall x\ y.[=]_{\tau_1}\ x\ y \longrightarrow [=]_{\tau_2}\ (f\ x)\ (g\ y)\big) = (f = g)$$

$$\vdots\ 3$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash [=]^{\bullet\circ}_{\tau_1} = [=]_{\tau_1}}\ \text{I.H.}}{\vdash [\doteq]^{\circ}_{\tau_1} = [=]_{\tau_1}}\ \text{defn}\bullet \qquad \dfrac{\dfrac{}{\vdash [=]^{\bullet\circ}_{\tau_2} = [=]_{\tau_2}}\ \text{I.H.}}{\vdash [\doteq]^{\circ}_{\tau_2} = [=]_{\tau_2}}\ \text{defn}\bullet \qquad \vdots\ 3}{\vdash \big(\forall x\ y.[=]_{\tau_1}\ x\ y \longrightarrow [=]_{\tau_2}\ (f\ x)\ (g\ y)\big) = (f = g)}\ \text{rewrite}^+}{\vdash \big(\forall x\ y.\, [\doteq]^{\circ}_{\tau_1}\ x\ y \longrightarrow [\doteq]^{\circ}_{\tau_2}\ (f\ x)\ (g\ y)\big) = (f = g)}\ \text{beta}}{\vdash \big(\forall x\ y.\, [\doteq]^{\circ}_{\tau_1}\ x\ y \longrightarrow [\doteq]^{\circ}_{\tau_2}\ (f\ x)\ (g\ y)\big)\ f\ g = (f = g)}\ \text{fun-ext}^+}{\vdash (\lambda f\ g.\ \forall x\ y.\, [\doteq]^{\circ}_{\tau_1}\ x\ y \longrightarrow [\doteq]^{\circ}_{\tau_2}\ (f\ x)\ (g\ y)) = [=]_{\tau_1 \Rightarrow \tau_2}}\ \text{defn}\circ}{\vdash (\lambda f\ g.\ \forall x\ y.\, x \doteq_{\tau_1} y \longrightarrow f\ x \doteq_{\tau_2} g\ y)^{\circ} = [=]_{\tau_1 \Rightarrow \tau_2}}\ \text{defn}\doteq}{\vdash [\doteq]^{\circ}_{\tau_1 \Rightarrow \tau_2} = [=]_{\tau_1 \Rightarrow \tau_2}}\ \text{defn}\bullet}{\vdash [=]^{\bullet\circ}_{\tau_1 \Rightarrow \tau_2} = [=]_{\tau_1 \Rightarrow \tau_2}}$$

$(f, g \text{ fresh})$

Subcase $[=]_{\Pi\alpha.\ \tau}$:

For the theorem $\vdash \mathcal{E}(\alpha)^{\circ}$ (1.1), the first three conditions follow from *trans*, *refl*, and *sym*. The final condition is shown as follows

$$\dfrac{\dfrac{\dfrac{}{\vdash [\doteq]^{\circ}_{\alpha \Rightarrow \alpha \Rightarrow o} = [=]_{\alpha \Rightarrow \alpha \Rightarrow o}}\ \text{I.H.} \qquad \dfrac{}{\vdash [\doteq]^{\circ}_{\alpha} = [=]_{\alpha}}\ \text{I.H.} \qquad \dfrac{}{\vdash [=]_{\alpha \Rightarrow \alpha \Rightarrow o}\ [=]_{\alpha}\ [=]_{\alpha}}\ \text{refl}}{\vdash [\doteq]^{\circ}_{\alpha \Rightarrow \alpha \Rightarrow o}\ [\doteq]^{\circ}_{\alpha}\ [\doteq]^{\circ}_{\alpha}}\ \text{rewrite}^+}{\vdash ([\doteq]_{\alpha} \doteq_{\alpha \Rightarrow \alpha \Rightarrow o} [\doteq]_{\alpha})^{\circ}}\ \text{defn}\circ$$

$$\dfrac{\dfrac{\dfrac{}{\forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:]) \vdash f\ [:\alpha:] =_{\Pi\alpha.\ \tau}\ g\ [:\alpha:]} \quad \vdash \mathcal{E}(\alpha)^\circ}{\forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:]) \vdash f\ [:\alpha:] =_{\Pi\alpha.\ \tau}\ g\ [:\alpha:]}\ {\to}\mathrm{E}}{\forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:]) \vdash f =_{\Pi\alpha.\ \tau}\ g}\ \text{tyfun-ext}\ (\alpha\ \text{fresh})$$

$$\dfrac{\dfrac{}{f =_{\Pi\alpha.\ \tau}\ g \vdash f =_{\Pi\alpha.\ \tau}\ g}\ \text{assm} \qquad \alpha\ \text{fresh}\ \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash (f\ [:\alpha:]) =_\tau (f\ [:\alpha:])}\ \text{refl}}{\mathcal{E}(\alpha)^\circ \vdash (f\ [:\alpha:]) =_\tau (f\ [:\alpha:])}\ \text{weaken}}{\vdash \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (f\ [:\alpha:])}\ {\to}\mathrm{I}}{\vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (f\ [:\alpha:])}\ \forall_{\mathsf{ty}}\mathrm{I}}{f =_{\Pi\alpha.\ \tau}\ g \vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:])}\ \text{rewrite}$$

$$\dfrac{\forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:]) =_o f =_{\Pi\alpha.\ \tau} g \qquad f =_{\Pi\alpha.\ \tau} g \vdash \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:])}{\vdash \left(\forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:])\right) =_o ([=]_{\Pi\alpha.\ \tau}\ f\ g)}\ \text{deduct-antisym}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash [=]^{\bullet\circ}_\tau = [=]_\tau} \quad \vdash \left(\forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [=]_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:])\right) =_o ([=]_{\Pi\alpha.\ \tau}\ f\ g)}{\vdash \left(\forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [\dot{=}]^\circ_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:])\right) =_o ([=]_{\Pi\alpha.\ \tau}\ f\ g)}\ \text{rewrite}^+}{\vdash \left(\lambda f\ g.\ \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha)^\circ \longrightarrow [\dot{=}]^\circ_\tau\ (f\ [:\alpha:])\ (g\ [:\alpha:])\right) = [=]_{\Pi\alpha.\ \tau}}\ \text{fun-ext}^+\ (f, g\ \text{fresh})}{\vdash \left(\lambda f\ g.\ \forall_{\mathsf{ty}}\alpha.\ \mathcal{E}(\alpha) \longrightarrow f\ [:\alpha:] \dot{=}_\tau g\ [:\alpha:]\right)^\circ = [=]_{\Pi\alpha.\ \tau}}\ \text{defn}\circ}{\vdash [\dot{=}]^\circ_{\Pi\alpha.\ \tau} = [=]_{\Pi\alpha.\ \tau}}\ \text{defn}[\dot{=}]}{\vdash [=]^{\bullet\circ}_{\Pi\alpha.\ \tau} = [=]_{\Pi\alpha.\ \tau}}\ \text{defn}\bullet$$

Subcase $[=]_\alpha$:

$$\dfrac{\dfrac{\dfrac{}{\vdash [=]_\alpha = [=]_\alpha}\ \text{refl}}{\vdash [\dot{=}]^\circ_\alpha = [=]_\alpha}\ \text{defn}\circ}{\vdash [=]^{\bullet\circ}_\alpha = [=]_\alpha}\ \text{defn}\bullet$$

□

C.2.2 Step 2: Subsystem Theorem

The second step of the non-triviality theorem is to show that, under the reverse translation, any proof in PHOL is a proof in $\text{PHOL}_{\xi\eta}$.

$$\frac{\Gamma \vdash_{\text{PHOL}} s}{\Gamma^\circ \vdash_{\text{PHOL}} s^\circ}\ \text{subsystem}$$

This step is obvious for any rule shared between PHOL and $\text{PHOL}_{\xi\eta}$; for rules $\forall$I, $\forall$E, $\forall_{\text{ty}}$I, and $\forall_{\text{ty}}$E, which are not primitive in $\text{PHOL}_{\xi\eta}$, note that we have proven these as derived rules in Section A.4.

C.2.3 Step 3: Removing $\mathcal{R}$ and $\mathcal{E}$

In $\text{PHOL}_{\xi\eta}$, we can show that $\mathcal{R}(\overline{x})^\circ$ ($\mathcal{R}\circ$) and $\mathcal{E}(\overline{\alpha})^\circ$ ($\mathcal{E}\circ$) are trivially true.

Proof. For $\mathcal{R}$, this follows immediately from the definition of $\circ$ and *refl*.

For $\mathcal{E}$, the first three conjuncts follow immediately from *trans*, *sym*, and *refl*. We have shown the final conjunct in Subsection C.2.1. □

C.2.4 Non-triviality Theorem

Finally, we can show the non-triviality of the translation as follows

$$\dfrac{\dfrac{}{\vdash^* \mathcal{E}(\Gamma^\bullet)^\circ, \mathcal{E}(s^\bullet)^\circ, \mathcal{R}(\Gamma^\bullet)^\circ, \mathcal{R}(s^\bullet)^\circ}\ \mathcal{R}\circ + \mathcal{E}\circ \quad \dfrac{\mathcal{E}(\Gamma^\bullet), \mathcal{E}(s^\bullet), \mathcal{R}(\Gamma^\bullet), \mathcal{R}(s^\bullet), \Gamma^\bullet \vdash_{\text{PHOL}} s^\bullet}{\mathcal{E}(\Gamma^\bullet)^\circ, \mathcal{E}(s^\bullet)^\circ, \mathcal{R}(\Gamma^\bullet)^\circ, \mathcal{R}(s^\bullet)^\circ, \Gamma^{\bullet\circ} \vdash_{\text{PHOL}_{\xi\eta}} s^{\bullet\circ}}\ \text{subsystem}}{\dfrac{\Gamma^{\bullet\circ} \vdash_{\text{PHOL}_{\xi\eta}} s^{\bullet\circ}}{\Gamma \vdash_{\text{PHOL}_{\xi\eta}} s}\ \text{rewrite + translate-reverse-eq-id}}\ \text{ante-subst}^*$$