Undecidability of Equality in the Free Locally Cartesian Closed Category

Simon Castellan, Pierre Clairambault, and Peter Dybjer

Abstract

We show that a version of Martin-Löf type theory with extensional identity, a unit type $\mathrm{N}_1, \Sigma, \Pi$, and a base type is a free category with families (supporting these type formers) both in a 1- and a 2-categorical sense. It follows that the underlying category of contexts is a free locally cartesian closed category in a 2-categorical sense because of a previously proved biequivalence. We then show that equality in this category is undecidable by reducing it to the undecidability of convertibility in combinatory logic.

1998 ACM Subject Classification F.4.1, F.3.2

Keywords and phrases Extensional type theory, locally cartesian closed categories, undecidability

Digital Object Identifier 10.4230/LIPIcs.TLCA.2015.x

1 Introduction

In previous work [4, 5] we showed the biequivalence of locally cartesian closed categories (lcccs) and the I, Σ, Π-fragment of extensional Martin-Löf type theory. More precisely, we showed the biequivalence of the following two 2-categories.

- The first has as *objects* lcccs, as *arrows* functors which preserve the lccc-structure (up to isomorphism), and as *2-cells* natural transformations.
- The second has as *objects* categories with families (cwfs) [7] which support extensional identity types (I-types), Σ-types, Π-types, and are *democratic*, as *arrows* pseudo cwf-morphisms (preserving cwf-structure up to isomorphism), and as *2-cells* pseudo cwf-transformations. A cwf is democratic iff there is an equivalence between its category of contexts and its category of closed types.

This result is a corrected version of a result by Seely [12] concerning the equivalence of the category of lcccs and the category of Martin-Löf type theories. Seely's paper did not address the coherence problem caused by the interpretation of substitution as pullbacks [6]. As Hofmann showed [8], this coherence problem can be solved by extending a construction of Bénabou [2]. Our biequivalence is based on this construction.

Cwfs are models of the most basic rules of dependent type theory; those dealing with substitution, assumption, and context formation, the rules which come before any rules for specific type formers. The distinguishing feature of cwfs, compared to other categorical notions of model of dependent types, is that they are formulated in a way which makes the connection with the ordinary syntactic formulation of dependent type theory transparent. They can be defined purely equationally [7] as a generalised algebraic theory (gat) [3], where each sort symbol corresponds to a judgment form, and each operator symbol corresponds to an inference rule in a variable free formulation of Martin-Löf's explicit substitution calculus for dependent type theory [10, 14].

Cwfs are not only models of dependent type theory, but also suggest an answer to the question what dependent type theory is as a mathematical object. Perhaps surprisingly, this is a non-trivial question, and Voevodsky has remarked that "a type system is not a

13th International Conference on Typed Lambda Calculi and Applications (TLCA'15).
Editor: Thorsten Altenkirch; pp. 1–22
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

mathematical notion". There are numerous variations of Martin-Löf type theory in the literature, even of the formulation of the most basic rules for dependent types. There are systems with explicit and implicit substitutions, variations in assumption, context formation, and substitution rules. There are formulations with de Bruijn indices and with ordinary named variables, etc. In fact, there are so many rules that most papers do not try to provide a complete list; and if you do try to list all of them how can you be sure that you haven't forgotten any? Nevertheless, there is a tacit assumption that most variations are equivalent and that a complete list of rules could be given if needed. However, from a mathematical point of view this is neither clear nor elegant.

To remedy this situation we suggest to define Martin-Löf type theory (and other dependent type theories) abstractly as the initial cwf (with extra structure). The category of cwfs and morphisms which preserve cwf-structure on the nose was defined by Dybjer [7]. We suggest that the correctness of a definition or an implementation of dependent type theory means that it gives rise to an initial object in this category of cwfs (with extra structure). Here we shall construct the initial object in this category explicitly in the simplest possible way following closely the definition of the generalised algebraic theory of cwfs. Note however that the notion of a generalised algebraic theory is itself based on dependent type theory, that is, on cwf-structure. So just defining the initial cwf as the generalised algebraic theory of cwfs would be circular. Instead we construct the initial cwf explicitly by giving grammar and inference rules which follow closely the operators of the gat of cwfs. However, we must also make equality reasoning explicit. To decrease the number of rules, we present a "per-style" system rather than an ordinary one. We will mutually define four partial equivalence relations (pers): for the judgments of context equality $\Gamma = \Gamma'$, substitution equality $\Delta \vdash \gamma = \gamma' : \Gamma$, type equality $\Gamma \vdash A = A'$, and term equality $\Gamma \vdash a = a' : A$. The ordinary judgments will be defined as the reflexive instances, for example, $\Gamma \vdash a : A$ will be defined as $\Gamma \vdash a = a : A$.

Our only optimisation is the elimination of some redundant arguments of operators. For example, the composition operator in the gat of cwfs has five arguments: three objects and two arrows. However, the three object arguments can be recovered from the arrows, and can hence be omitted. This method is also used in *D-systems*, the essentially algebraic formulation of cwfs by Voevodsky.

The goal of the present paper is to prove the undecidability of equality in the free lccc. To this end we extend our formal system for cwfs with rules for extensional I-types, $\mathrm{N}_1, \Sigma, \Pi$, and a base type. Now we want to show that this yields a free lccc on one object, by appealing to our biequivalence theorem. (Since the empty context corresponds to the unit type N_1 and context extension to Σ, it follows that our free cwf is democratic.) However, it does not suffice to show that we get a free cwf in the 1-category of cwfs and strict cwf-morphism, but we must show that it is also free ("bifree") in the 2-category of cwfs and pseudo cwf-morphisms. Although informally straightforward, this proof is technically more involved because of the complexity of the notion of pseudo cwf-morphism.

Once we have constructed the free lccc (as a cwf-formulation of Martin-Löf type theory with extensional I-types, $\mathrm{N}_1, \Sigma, \Pi$, and one base type) we will be able to prove undecidability. It is a well-known folklore result that extensional Martin-Löf type theory with one universe has undecidable equality, and we only need to show that a similar construction can be made without a universe, provided we have a base type. We do this by encoding untyped combinatory logic as a context, and use the undecidability of equality in this theory.

Related work. Palmgren and Vickers [11] show how to construct free models of essentially algebraic theories in general. We could use this result to build a free cwf, but this only shows freeness in the 1-categorical sense. We also think that the explicit construction of the free (and bifree) cwf is interesting in its own right.

Plan. In Section 2 we prove a few undecidability theorems, including the undecidability of equality in Martin-Löf type theory with extensional I-types, $\mathrm{N}_1, \Sigma, \Pi$, and one base type. In Section 3 we construct a free cwf on one base type. We show that it is free both in a 1-categorical sense (where arrows preserve cwf-structure on the nose) and in a 2-categorical sense (where arrows preserve cwf-structure up to isomorphism). In Section 4 we construct a free cwf with extensional identity types, $\mathrm{N}_1, \Sigma, \Pi$, and one base type. We then use the biequivalence result to conclude that this yields a free lccc in a 2-categorical sense.

2 Undecidability in Martin-Löf type theory

Like any other single-sorted first order equational theory, combinatory logic can be encoded as a context in Martin-Löf type theory with I-types, Π-types, and a base type o. The context Γ_{CL} for combinatory logic is the following:

$$\begin{array}{rclrcl} k & : & o, & ax_k & : & \Pi xy : o.\mathrm{I}(o, k.x.y, x), \\ s & : & o, & ax_s & : & \Pi xyz : o.\mathrm{I}(o, s.x.y.z, x.z.(y.z)) \\ . & : & o \to o \to o, & & & \end{array}$$

Here we have used the left-associative binary infix symbol "." for application. Note that $k, s, ., ax_k, ax_s$ are all variables.

▶ **Theorem 1.** *Type-inhabitation in Martin-Löf type theory with (intensional or extensional) identity-types,* Π*-types and a base type is undecidable.*

This follows from the undecidability of convertibility in combinatory logic, because the type

$$\Gamma_{\mathrm{CL}} \vdash \mathrm{I}(o, M, M')$$

is inhabited iff the closed combinatory terms M and M' are convertible. Clearly, if the combinatory terms are convertible, it can be formalized in this fragment of type theory. For the other direction we build a model of the context Γ_{CL} where o is interpreted as the set of combinatory terms modulo convertibility.

▶ **Theorem 2.** *Judgmental equality in Martin-Löf type theory with extensional identity-types,* Π*-types and a base type is undecidable.*

With extensional identity types [9] the above identity type is inhabited iff the corresponding equality judgment is valid:

$$\Gamma_{\mathrm{CL}} \vdash M = M' : o$$

This theorem also holds if we add N_1 and Σ-types to the theory. The remainder of the paper will show that the category of contexts for the resulting fragment of Martin-Löf type theory is free ("bifree") in the 2-category of lcccs (Theorem 20). Our main result follows:

▶ **Theorem 3.** *Equality of arrows in the bifree lccc on one object is undecidable.*

We would like to remark that the following folklore theorem can be proved in the same way.

▶ **Theorem 4.** *Judgmental equality in Martin-Löf type theory with extensional identity-types,* Π*-types and a universe* U *is undecidable.*

If we have a universe we can instead work in the context

$$\begin{array}{rclrcl} X & : & \mathrm{U} & . & : & X \to X \to X, \\ k & : & X, & ax_k & : & \Pi xy : X.\mathrm{I}(X, k.x.y, x), \\ s & : & X, & ax_s & : & \Pi xyz : X.\mathrm{I}(X, s.x.y.z, x.z.(y.z)) \end{array}$$

and prove undecidability for this theory (without a base type) in the same way as above.

Note that we don't need any closure properties at all for U – only the ability to quantify over small types. Hence we prove a slightly stronger theorem than the folklore theorem which assumes that U is closed under function types, and then uses the context

$$X \;:\; \mathrm{U}, \qquad x \;:\; \mathrm{I}(\mathrm{U}, X, X \to X)$$

so that X is a model of the untyped lambda calculus.

3 A free category with families

In this section we define a free cwf syntactically, as a *term model* consisting of derivable contexts, substitutions, types and terms modulo derivable equality. To this end we give a syntax and inference rules for a cwf-calculus, that is, a variable free explicit substitution calculus for dependent type theory.

We first prove that this calculus yields a free cwf in the category where morphisms preserve cwf-structure on the nose. The free cwf on one object is a rather degenerate structure, since there are no non-trivial dependent types. However, we have nevertheless chosen to present this part of the construction separately. Cwfs model the common core of dependent type theory, including all generalised algebraic theories, pure type systems [1], and fragments of Martin-Löf type theory. The construction of a free pure cwf is thus the common basis for constructing free and initial cwfs with appropriate extra structure for modelling specific dependent type theories.

In Section 4 we prove that our free cwf is also bifree. We then extend this result to cwfs supporting N_1, Σ, and Π-types. By our biequivalence result [4, 5] it also yields a bifree lccc.

3.1 The 2-category of categories with families

The 1-category of cwfs and morphisms which preserve cwf-structure on the nose was defined in [7]. The 2-category of cwfs and pseudo-morphisms which preserve cwf-structure up to isomorphism was defined in [4, 5]. Here we only give an outline.

▶ **Definition 5** (Category with families)**.** A cwf $\mathcal{C}$ is a pair $(\mathcal{C}, T)$ of a category $\mathcal{C}$ and a functor $T : \mathcal{C}^{\mathrm{op}} \to \mathbf{Fam}$ where $\mathbf{Fam}$ is the category of families of sets. We write $\mathrm{Ctx}_{\mathcal{C}} = |\mathcal{C}|$ and $\mathrm{Sub}_{\mathcal{C}}(\Delta, \Gamma) = \mathrm{Hom}_{\mathcal{C}}(\Delta, \Gamma) = \Delta \to C$. For $\Gamma \in \mathrm{Ctx}_{\mathcal{C}}$ we write $T\Gamma = (\mathrm{Tm}_{\mathcal{C}}(\Gamma, A))_{A \in \mathrm{Ty}_{\mathcal{C}}\Gamma}$. The functorial action of T on a type A is written $A\_$ or $A[\_]$ (depending on which is more readable): if $\gamma : \mathrm{Sub}_{\mathcal{C}}(\Gamma, \Delta)$ and $A \in \mathrm{Ty}_{\mathcal{C}}(\Delta)$, $A[\gamma] \in \mathrm{Ty}_{\mathcal{C}}(\Gamma)$. Similarly if $a \in \mathrm{Tm}_{\mathcal{C}}(\Delta, A)$, we write $a[\gamma] \in \mathrm{Tm}_{\mathcal{C}}(\Gamma, A[\gamma])$ (or $a\gamma$) for the functorial action of T on a.

We assume that $\mathcal{C}$ has a terminal object 1. Moreover we assume that for each $\Gamma \in \mathrm{Ctx}_{\mathcal{C}}$ and $A \in \mathrm{Ty}_{\mathcal{C}}(\Gamma)$ there exists $\Gamma.A \in \mathrm{Ctx}_{\mathcal{C}}$ with a map $\mathtt{p}_A : \mathrm{Sub}_{\mathcal{C}}(\Gamma.A, \Gamma)$ and a term $\mathtt{q}_A \in \mathrm{Tm}_{\mathcal{C}}(\Gamma.A, A[\mathtt{p}_A])$, such that for every pair $\gamma : \mathrm{Sub}_{\mathcal{C}}(\Delta, \Gamma)$ and $a \in \mathrm{Tm}_{\mathcal{C}}(\Delta, A[\gamma])$ there exists a unique map $\langle \gamma, a \rangle : \mathrm{Sub}_{\mathcal{C}}(\Delta, \Gamma.A)$ such that $\mathtt{p}_A \circ \langle \gamma, a \rangle = \gamma$ and $\mathtt{q}_A[\langle \gamma, a \rangle] = a$.

Note that with the notation $\mathrm{Ty}_{\mathcal{C}}$ and $\mathrm{Tm}_{\mathcal{C}}$ there is no need to explicitly mention the functor T when working with the category with families, and we will often omit it. Given a substitution $\gamma : \Gamma \to \Delta$, and $A \in \mathrm{Ty}_{\mathcal{C}}(\Delta)$, we write $\gamma \uparrow A$ or γ^+ (when A can be inferred from the context) for the lifting of γ to A: $\langle \gamma \circ \mathtt{p}, \mathtt{q}\rangle : \Gamma.A\gamma \to \Delta.A$.

The indexed category. In [4, 5] it is shown that any cwf $\mathcal{C}$ induces a functor $\mathbf{T} : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$ assigning to each context Γ the category whose objects are types in $\mathrm{Ty}_{\mathcal{C}}(\Gamma)$ and morphisms from A to B are substitutions $\gamma : \Gamma.A \to \Gamma.B$ such that $\mathtt{p} \circ \gamma = \gamma$. (They are in bijection with terms of type $\Gamma \cdot A \vdash B\mathtt{p}$. Any morphism γ in $\mathbf{T}\Gamma$ from a type A to B induces a function on terms of that type written $\{\gamma\} : \mathrm{Tm}_{\mathcal{C}}(\Gamma, A) \to \mathrm{Tm}_{\mathcal{C}}(\Gamma, B)$ defined by $\{\gamma\}(a) = \mathtt{q}[\gamma \circ \langle \mathtt{id}, a\rangle]$. We will write $\theta : A \cong_\Gamma B$ for an isomorphism in $\mathbf{T}\Gamma$.

The functorial action is given by $\mathbf{T}(\gamma)(\varphi) = \langle \mathtt{p}, \mathtt{q}[\varphi \circ \gamma \uparrow A]\rangle : \Gamma.A[\gamma] \to \Gamma.B[\gamma]$, from which we deduce the action on terms $\{\mathbf{T}(\gamma)(\varphi)\}(a) = \mathtt{q}[\varphi \circ \langle \gamma, a\rangle]$.

▶ **Definition 6** (Pseudo cwf-morphisms)**.** A pseudo-cwf morphism from a cwf $(\mathcal{C}, T)$ to a cwf $(\mathcal{D}, T')$ is a pair (F, σ) where $F : \mathcal{C} \to \mathcal{D}$ is a functor and for each $\Gamma \in \mathcal{C}$, σ_Γ is a **Fam**-morphism from $T\Gamma$ to $T'F\Gamma$ preserving the structure up to isomorphism. For example, there are isomorphisms

- $\rho_{\Gamma,A} : F(\Gamma.A) \cong F\Gamma.FA$
- $\theta_{A,\gamma} : F\Gamma.FA[F\gamma] \cong F\Gamma.F(A\gamma)$ for $\Gamma \vdash \gamma : \Delta$.

satisfying some coherence diagrams, see [5] for the complete definition.

As σ_Γ is a **Fam**-morphism from $(\mathrm{Tm}_{\mathcal{C}}(\Gamma, A))_{A \in \mathrm{Ty}_{\mathcal{C}}(\Gamma)}$ to $(\mathrm{Tm}_{\mathcal{D}}(F\Gamma, B))_{B \in \mathrm{Ty}_{\mathcal{D}}(F\Gamma)}$, we will write FA for the image of A by $\mathrm{Ty}_{\mathcal{C}}(\Gamma) \to \mathrm{Ty}_{\mathcal{D}}(F\Gamma)$ induced by σ_Γ and Fa for the image of $\Gamma \vdash a : A$ through $\mathrm{Tm}_{\mathcal{C}}(\Gamma, A) \to \mathrm{Tm}_{\mathcal{D}}(F\Gamma, FA)$ induced by σ_Γ.

A pseudo cwf-morphism is strict whenever $\theta_{A,\gamma}$ and $\rho_{\Gamma,A}$ are both identities and $F1 = 1$. Cwfs and strict cwf-morphisms form a category $\mathbf{CwF}_s$.

▶ **Definition 7** (Pseudo cwf-transformation)**.** A pseudo cwf-transformation between functors (F, σ) and (G, τ) is a pair (φ, ψ) where $\varphi : F \Rightarrow G$ is a natural transformation, and for each $\Gamma \in \mathcal{C}$ and $A \in \mathrm{Ty}_{\mathcal{C}}(\Gamma)$ $\psi_{\Gamma,A}$ is a type isomorphism $FA \cong GA[\varphi_\Gamma]$ satisfying:

$$\varphi_{\Gamma.A} = F(\Gamma.A) \xrightarrow{\rho_F} F\Gamma.FA \xrightarrow{\psi_{\Gamma,A}} F\Gamma.GA[\varphi_\Gamma] \xrightarrow{\varphi_\Gamma^+} G\Gamma.GA \xrightarrow{\rho_G^{-1}} G(\Gamma.A)$$

We will write **CwF** for the resulting 2-category.

3.2 Syntax and inference rules for the free category with families

3.2.1 Raw terms

In this section we define the syntax and inference rules for a minimal dependent type theory with one base type o. This theory is closely related to the generalised algebraic theory of cwfs [7], but here we define it as a usual logical system with a grammar and a collection of inference rules. The grammar has four syntactic categories: contexts `Ctx`, substitutions `Sub`, types `Ty` and terms `Tm`:

$$\begin{array}{lclclcl} \Gamma & ::= & 1 \mid \Gamma.A & \qquad & A & ::= & o_\Gamma \mid A\,\gamma \\ \gamma & ::= & \gamma \circ \gamma \mid \mathtt{id}_\Gamma \mid \langle\rangle_\Gamma \mid \mathtt{p}_A \mid \langle \gamma, a\rangle_A & & a & ::= & a\,\gamma \mid \mathtt{q}_A \end{array}$$

These terms have as few annotations as possible, only what is needed to recover the domain and codomain of a substitution, the context of a type, and the type of a term:

$$\begin{aligned}
\mathrm{dom}(\gamma \circ \gamma') &= \mathrm{dom}(\gamma') & \mathrm{cod}(\gamma \circ \gamma') &= \mathrm{cod}(\gamma)\\
\mathrm{dom}(\mathtt{id}_\Gamma) &= \Gamma & \mathrm{cod}(\mathtt{id}_\Gamma) &= \Gamma\\
\mathrm{dom}(\langle\rangle_\Gamma) &= \Gamma & \mathrm{cod}(\langle\rangle_\Gamma) &= 1\\
\mathrm{dom}(\mathtt{p}_A) &= \text{ctx-of}(A).A & \mathrm{cod}(\mathtt{p}_A) &= \text{ctx-of}(A)\\
\mathrm{dom}(\langle\gamma, a\rangle_A) &= \mathrm{dom}(\gamma) & \mathrm{cod}(\langle\gamma, a\rangle_A) &= \mathrm{cod}(\gamma).A
\end{aligned}$$

$$\begin{aligned}
\text{ctx-of}(o_\Gamma) &= \Gamma & \text{type-of}(a\,\gamma) &= (\text{type-of}(a))\,\gamma\\
\text{ctx-of}(A\,\gamma) &= \mathrm{cod}(\gamma) & \text{type-of}(\mathtt{q}_A) &= A\,\mathtt{p}_A
\end{aligned}$$

These functions will be used in the freeness proof.

3.2.2 Inference rules

We simultaneously inductively define four families of partial equivalence relations (pers) for the four forms of equality judgment:

$$\Gamma = \Gamma' \vdash \qquad \Gamma \vdash A = A' \qquad \Delta \vdash \gamma = \gamma' : \Gamma \qquad \Gamma \vdash a = a' : A$$

In the inference rules which generate these pers we will use the following abbreviations for the basic judgment forms: $\Gamma \vdash$ abbreviates $\Gamma = \Gamma \vdash$, $\Gamma \vdash A$ abbreviates $\Gamma \vdash A = A$, $\Delta \vdash \gamma : \Gamma$ abbreviates $\Delta \vdash \gamma = \gamma : \Gamma$, and $\Gamma \vdash a : A$ abbreviates $\Gamma \vdash a = a : A$

Per-rules for the four forms of judgments:

$$\frac{\Gamma = \Gamma' \vdash \quad \Gamma' = \Gamma'' \vdash}{\Gamma = \Gamma'' \vdash} \qquad \frac{\Gamma = \Gamma' \vdash}{\Gamma' = \Gamma \vdash} \qquad \frac{\Delta \vdash \gamma = \gamma' : \Gamma \quad \Delta \vdash \gamma' = \gamma'' : \Gamma}{\Delta \vdash \gamma = \gamma'' : \Gamma} \qquad \frac{\Delta \vdash \gamma = \gamma' : \Gamma}{\Delta \vdash \gamma' = \gamma : \Gamma}$$

$$\frac{\Gamma \vdash A = A' \quad \Gamma \vdash A' = A''}{\Gamma \vdash A = A''} \qquad \frac{\Gamma \vdash A = A'}{\Gamma \vdash A' = A} \qquad \frac{\Gamma \vdash a = a' : A \quad \Gamma \vdash a' = a'' : A}{\Gamma \vdash a = a'' : A}$$

$$\frac{\Gamma \vdash a = a' : A}{\Gamma \vdash a' = a : A}$$

Preservation rules for judgments:

$$\frac{\Gamma = \Gamma' \vdash \quad \Delta = \Delta' \vdash \quad \Gamma \vdash \gamma = \gamma' : \Delta}{\Gamma' \vdash \gamma = \gamma' : \Delta'} \qquad \frac{\Gamma = \Gamma' \vdash \quad \Gamma \vdash A = A'}{\Gamma' \vdash A = A'}$$

$$\frac{\Gamma = \Gamma' \vdash \quad \Gamma \vdash A = A' \quad \Gamma \vdash a = a' : A}{\Gamma' \vdash a = a' : A'}$$

Congruence rules for operators:

$$\frac{\Gamma \vdash \delta = \delta' : \Delta \quad \Delta \vdash \gamma = \gamma' : \Theta}{\Gamma \vdash \delta \circ \gamma = \delta' \circ \gamma' : \Theta} \qquad \frac{\Gamma = \Gamma' \ \vdash}{\Gamma \ \vdash \ \mathtt{id}_\Gamma = \mathtt{id}_{\Gamma'} : \Gamma} \qquad \frac{\Gamma \vdash A = A' \quad \Delta \vdash \gamma = \gamma' : \Gamma}{\Delta' \vdash A\,\gamma = A'\,\gamma'}$$

$$\frac{\Gamma \vdash a = a' : A \quad \Delta \vdash \gamma = \gamma' : \Gamma}{\Delta' \vdash a\,\gamma = a'\,\gamma' : A'\,\gamma'} \qquad \frac{}{1 = 1 \vdash} \qquad \frac{\Gamma = \Gamma' \ \vdash}{\Gamma \vdash \langle\rangle_\Gamma = \langle\rangle_{\Gamma'} : 1} \qquad \frac{\Gamma = \Gamma' \vdash \quad \Gamma \vdash A = A'}{\Gamma.A = \Gamma'.A' \vdash}$$

$$\frac{\Gamma \vdash A = A'}{\Gamma.A \vdash \mathtt{p}_A = \mathtt{p}_{A'} : \Gamma} \qquad \frac{\Gamma \vdash A = A'}{\Gamma.A \vdash \mathtt{q}_A = \mathtt{q}_{A'} : A\,\mathtt{p}_A}$$

$$\frac{\Gamma \vdash A = A' \quad \Delta \vdash \gamma = \gamma' : \Gamma \quad \Delta \vdash a = a' : A\,\gamma}{\Delta \vdash \langle \gamma, a \rangle_A = \langle \gamma', a' \rangle_{A'} : \Gamma.A}$$

Conversion rules:

$$\frac{}{(\theta \circ \delta) \circ \gamma = \theta \circ (\delta \circ \gamma)} \qquad \frac{\Gamma \ \vdash \ \gamma : \Delta}{\Gamma \vdash \gamma = \mathtt{id}_\Delta \circ \gamma : \Delta} \qquad \frac{\Gamma \ \vdash \ \gamma : \Delta}{\Gamma \vdash \gamma = \gamma \circ \mathtt{id}_\Gamma : \Delta}$$

$$\frac{\Gamma \vdash A \quad \Delta \vdash \gamma : \Gamma \quad \Theta \vdash \delta : \Delta}{\Theta \vdash A\,(\delta \circ \gamma) = (A\,\delta)\,\gamma} \qquad \frac{\Gamma \vdash A}{\Gamma \vdash A\,\mathtt{id}_\Gamma = A} \qquad \frac{\Gamma \vdash a : A \quad \Delta \vdash \gamma : \Gamma \quad \Gamma \vdash \delta : \Delta}{\Theta \vdash a\,(\delta \circ \gamma) = (a\,\delta)\,\gamma : (A\,\delta)\,\gamma}$$

$$\frac{\Gamma \vdash a : A}{\Theta \vdash a\,\mathtt{id}_\Gamma = a : A} \qquad \frac{\Gamma \vdash \gamma : 1}{\Gamma \vdash \gamma = \langle\rangle_\Gamma : 1} \qquad \frac{\Gamma \vdash A \quad \Delta \vdash \gamma : \Gamma \quad \Delta \vdash a : A\,\gamma}{\Delta \vdash \mathtt{p}_A \circ \langle \gamma, a \rangle_A = \gamma : \Gamma}$$

$$\frac{\Gamma \vdash A \quad \Delta \vdash \gamma : \Gamma \quad \Delta \vdash a : A\,\gamma}{\Delta \vdash \mathtt{q}_A\,\langle \gamma, a \rangle_A = a : A\,\gamma} \qquad \frac{\Delta \vdash \gamma : \Gamma.A}{\Delta \vdash \gamma = \langle \mathtt{p}_A \circ \gamma, \mathtt{q}_A\,\gamma \rangle_A : \Gamma.A}$$

Rule for the base type:

$$\frac{}{1 \vdash o = o}$$

3.2.3 The syntactic cwf $\mathcal{T}$

We can now define a term model as the syntactic cwf obtained by the well-formed contexts, etc, modulo judgmental equality:

▶ **Definition 8.** The term model $\mathcal{T}$ is given by:

- $\mathrm{Ctx}_\mathcal{T} = \{\Gamma \mid \Gamma \vdash \}/{=^c}$, where $\Gamma =^c \Gamma'$ if $\Gamma = \Gamma' \vdash$ is derivable.
- $\mathrm{Sub}_\mathcal{T}([\Gamma], [\Delta]) = \{\gamma \mid \Gamma \vdash \gamma : \Delta\}/{=^\Gamma_\Delta}$ where $\gamma =^\Gamma_\Delta \gamma'$ iff $\Gamma \vdash \gamma = \gamma' : \Delta$ is derivable. Note that this makes sense since it only depends on the equivalence class of Γ and morphisms and morphism equality are preserved by object equality.
- $\mathrm{Ty}_\mathcal{T}([\Gamma]) = \{A \mid \Gamma \vdash A\}/ =^\Gamma$ where $A =^\Gamma B$ if $\Gamma \vdash A = B$. Again this is a well-defined for the same reason.
- $\mathrm{Tm}_\mathcal{T}([\Gamma], [A]) = \{a \mid \Gamma \vdash a : A\}/ =^\Gamma_A$ where $a =^\Gamma_A a'$ if $\Gamma \vdash a = a' : A$.

The cwf-operations on $\mathcal{T}$ can now be defined in a straightforward way. For example, if $\Delta \vdash \theta : \Theta$, $\Gamma \vdash \delta : \Delta$, we define $[\theta] \circ_\mathcal{T} [\delta] = [\theta \circ \delta]$, which is well-defined since composition preserves equality.

3.3 Freeness of $\mathcal{T}$

We shall now show that $\mathcal{T}$ is the free cwf on one base type, in the sense that given a cwf $\mathcal{C}$ and a type $A \in \mathrm{Ty}_\mathcal{C}(1)$, there exists a unique strict cwf morphism $[\![-]\!]_\mathcal{C} : \mathcal{T} \to \mathcal{C}$ such that $[\![o]\!] = A$. This can be defined by first defining a partial function for each sort of raw terms (where `Ctx` denotes the set of raw contexts, `Sub` the set of raw substitutions, and so on defined by the grammar of Section 3.2.1), cf Streicher [13].

$$\begin{array}{rcl} [\![-]\!] & : & \texttt{Ctx} \to \mathrm{Ctx}_{\mathcal{C}} \\ [\![-]\!]_{\Gamma,\Delta} & : & \texttt{Sub} \to \mathrm{Sub}_{\mathcal{C}}([\![\Gamma]\!],[\![\Delta]\!]) \\ [\![-]\!]_{\Gamma} & : & \texttt{Ty} \to \mathrm{Ty}_{\mathcal{C}}([\![\Gamma]\!]) \\ [\![-]\!]_{\Gamma,A} & : & \texttt{Tm} \to \mathrm{Tm}_{\mathcal{C}}([\![\Gamma]\!],[\![A]\!]_{\Gamma}) \end{array}$$

These functions are defined by mutual induction on the structure of raw terms:

$$\begin{array}{rclrcl} [\![1]\!] &=& 1_{\mathcal{C}} & [\![\Gamma.A]\!] &=& [\![\Gamma]\!].{}_{\mathcal{C}}[\![A]\!] \\ [\![\gamma' \circ \gamma]\!]_{\Gamma,\Theta} &=& [\![\gamma']\!]_{\Delta,\Theta} \circ_{\mathcal{C}} [\![\gamma]\!]_{\Gamma,\Delta} & [\![\mathtt{id}_{\Gamma}]\!]_{\Gamma,\Gamma} &=& \mathrm{id}_{\mathcal{C}[\![\Gamma]\!]} \\ [\![\langle\gamma,a\rangle_A]\!]_{\Gamma,\Delta.A} &=& \langle[\![\gamma]\!]_{\Gamma,\Delta},[\![a]\!]_{\Gamma,A[\gamma]}\rangle & [\![\langle\rangle_{\Gamma}]\!]_{\Gamma,1} &=& \langle\rangle_{\Gamma} \\ [\![A\,\gamma]\!]_{\Gamma} &=& [\![A]\!]_{\Delta}\,[\![\gamma]\!]_{\Gamma,\Delta} & [\![a\,\gamma]\!]_{\Gamma,A[\gamma]} &=& [\![a]\!]_{\Delta,A}\,[\![\gamma]\!]_{\Gamma,\Delta} \\ [\![\mathrm{p}_A]\!]_{\Gamma.A,\Gamma} &=& \mathtt{p}_A & [\![\mathtt{q}_A]\!]_{\Gamma.A,A[p]} &=& \mathtt{q}_{,A} \\ [\![o]\!]_1 &=& A & & & \end{array}$$

Note that $\Delta = \mathrm{dom}(\gamma') = \mathrm{cod}(\gamma)$ in the equation for $\circ$, etc.

We then prove by induction on the inference rules that

▶ **Lemma 9.**
- *If $\Gamma = \Gamma' \vdash$, then $[\![\Gamma]\!] = [\![\Gamma']\!]$ and both are defined.*
- *If $\Gamma \vdash \gamma = \gamma' : \Delta$, then $[\![\gamma]\!]_{\Gamma,\Delta} = [\![\gamma']\!]_{\Gamma,\Delta}$ and both are defined.*
- *If $\Gamma \vdash A = A'$, then $[\![A]\!]_{\Gamma} = [\![A']\!]_{\Gamma}$ and both are defined.*
- *If $\Gamma \vdash a = a' : A$, then $[\![a]\!]_{\Gamma,A} = [\![a']\!]_{\Gamma,A}$ and both are defined.*

Hence the partial interpretation lifts to the quotient of syntax by judgmental equality:

$$\begin{array}{rcl} \overline{[\![-]\!]} & : & \mathrm{Ctx}_{\mathcal{T}} \to \mathrm{Ctx}_{\mathcal{C}} \\ \overline{[\![-]\!]}_{[\Gamma],[\Delta]} & : & \mathrm{Sub}_{\mathcal{T}}([\Gamma],[\Delta]) \to \mathrm{Sub}_{\mathcal{C}}(\overline{[\![[\Gamma]]\!]},\overline{[\![[\Delta]]\!]}) \\ \overline{[\![-]\!]}_{[\Gamma]} & : & \mathrm{Ty}_{\mathcal{T}}([\Gamma]) \to \mathrm{Ty}_{\mathcal{C}}(\overline{[\![[\Gamma]]\!]}) \\ \overline{[\![-]\!]}_{[\Gamma],[A]} & : & \mathrm{Tm}_{\mathcal{T}}([\Gamma],[A]) \to \mathrm{Tm}_{\mathcal{C}}(\overline{[\![[\Gamma]]\!]},\overline{[\![[A]]\!]}_{[\Gamma]}) \end{array}$$

This defines a strict cwf morphism $\mathcal{T} \to \mathcal{C}$ which sends o to A. It is easy to check that it is the unique such strict cwf-morphism. Hence we have proved

▶ **Theorem 10.** *$\mathcal{T}$ is the free cwf on one object, that is, for every other cwf $\mathcal{C}$ and $A \in \mathrm{Ty}_{\mathcal{C}}(1)$ there exists a unique strict cwf morphism $\mathcal{T} \to \mathcal{C}$ sending o_1 to A.*

3.4 Bifreeness of $\mathcal{T}$

We recall that an object I is bi-initial in a 2-category iff for any object A there exists an arrow $I \to A$ and for any two arrows $f, g : I \to A$ there exists a unique 2-cell $\theta : f \dot{\to} g$. It follows that θ is invertible. It also follows that bi-initial objects are equivalent.

▶ **Definition 11.** A cwf $\mathcal{C}$ is bifree on one base type iff it is bi-initial in the 2-category $\mathbf{CwF}^o$:
- *Objects*: A pair $(\mathcal{C}, o_{\mathcal{C}})$ of a CwF and a chosen $o_{\mathcal{C}} \in \mathrm{Ty}_{\mathcal{C}}(1)$,
- *1-cells from $(\mathcal{C}, o_{\mathcal{C}})$ to $(\mathcal{D}, o_{\mathcal{D}})$*: pseudo cwf-morphisms $F : \mathcal{C} \to \mathcal{D}$ such that there exists $\varphi_F : F(o_{\mathcal{C}}) \cong o_{\mathcal{D}}$ in $Ty_{\mathcal{D}}(1)$,

- *2-cells between from F to G with type* $(\mathcal{C}, o_\mathcal{C}) \to (\mathcal{D}, o_\mathcal{D})$: pseudo cwf-transformations (φ, ψ) from F to G satisfying $\psi_{o_\mathcal{C}} = \varphi_G^{-1} \circ \alpha_F : F(o_\mathcal{C}) \cong G(o_\mathcal{C})$.

▶ **Theorem 12.** *$\mathcal{T}$ is a bifree cwf on one base type.*

We have showed that for every cwf $\mathcal{C}$, $A \in \mathrm{Ty}_\mathcal{C}(1)$, the interpretation $\overline{[\![-]\!]}$ is a strict cwf-morphism mapping o to A. Hence it is a morphism in $\mathbf{CwF}^o$. It remains to show that for any other morphism $F : \mathcal{T} \to \mathcal{C}$ in $\mathbf{CwF}^o$, there is a unique 2-cell (pseudo cwf-transformation) $(\varphi, \psi) : \overline{[\![-]\!]} \cong F$, which happens to be an isomorphism. This asymetric version of bi-initiality is equivalent to that given below because the 2-cell we build is an isomorphism.

Existence of (φ, ψ). We construct (φ, ψ) by induction on the inference rules and simultaneously prove their naturality properties:

- If $\Gamma = \Gamma' \vdash$, then there exist $\varphi_\Gamma = \varphi_{\Gamma'} : \overline{[\![\Gamma]\!]} \cong F[\Gamma]$[1]
- If $\Gamma \vdash A = A'$, then there exist $\psi_A = \psi_{A'} : \overline{[\![\Gamma.A]\!]} \cong_{\overline{[\![\Gamma]\!]}} \overline{[\![\Gamma]\!]}.FA[\varphi_\Gamma]$ in $\mathbf{T}\overline{[\![\Gamma]\!]}$.
- If $\Gamma \vdash \gamma = \gamma' : \Delta$, then $F\gamma \circ \varphi_\Gamma = \varphi_\Delta \circ \overline{[\![\gamma]\!]}$
- If $\Gamma \vdash a = a' : A$, then $\{\psi_A\}(\overline{[\![a]\!]}) = Fa[\varphi_\Gamma]$

Since it also follows that $\varphi_{\Gamma.A} = \rho^{-1} \circ \varphi_\Gamma^+ \circ \psi_A$ we conclude that (φ, ψ) is a pseudo cwf-transformation. For space reasons we only present the proofs of the first two items and refer the reader to the long version of the paper [**?**] for the other two.

Empty context. F preserves terminal objects, thus we let $\phi_1 : \overline{[\![1]\!]} = 1_\mathcal{C} \cong F1$.

Context extension. By induction, we have $\psi_A : \overline{[\![A]\!]} \cong_\Gamma FA(\varphi_\Gamma)$ [2]. We define $\varphi_{\Gamma.A}$ as the following composition of isomorphisms:

$$\varphi_{\Gamma.A} = \overline{[\![\Gamma.A]\!]} \xrightarrow{\psi_A} \overline{[\![\Gamma]\!]}.FA(\varphi_\Gamma) \xrightarrow{\langle \varphi_\Gamma, \mathtt{q} \rangle} F\Gamma.FA \xrightarrow{\rho^{-1}_{\Gamma,A}} F(\Gamma.A)$$

Type substitution. Let $\Gamma \vdash \gamma : \Delta$ and $\Delta \vdash A$. By induction we get $\varphi_\Delta \circ \overline{[\![\gamma]\!]} = F\gamma \circ \varphi_\Gamma$ and $\psi_A : \overline{[\![A]\!]} \cong_\Delta FA(\varphi_\Delta)$. Since $\mathbf{T}$ is a functor, $\mathbf{T}\gamma$ is a functor from $\mathbf{T}\Delta$ to $\mathbf{T}\Gamma$ thus,

$$\mathbf{T}(\overline{[\![\gamma]\!]})(\psi_A) : \overline{[\![A\gamma]\!]} \cong_\Gamma FA[\varphi_\Delta \circ \gamma] = FA[F\gamma][\varphi_\Gamma]$$

by induction hypothesis on γ. So we define

$$\psi_{A\gamma} = \mathbf{T}(\varphi_\Gamma)(\theta_{A,\gamma}) \circ \mathbf{T}(\overline{[\![\gamma]\!]})(\psi_A) : \overline{[\![A\gamma]\!]} \cong_{\overline{[\![\Gamma]\!]}} F(A\gamma)[\varphi_\Gamma]$$

Unfolding the definition, this yields the following term: Using the previous case we can get a simpler equation for $\varphi_{\Gamma.A\gamma}$:

$$\varphi_{\Gamma.A\gamma} = \langle \varphi_\Gamma \circ \mathtt{p}, \mathtt{q}[\rho \circ \varphi_{\Delta\cdot A} \circ \gamma \uparrow A] \rangle : \overline{[\![\Gamma.A[\gamma]]\!]} \to F(\Gamma.A\gamma)$$

Base type. By definition, F comes equipped with $\alpha_F : \overline{[\![o]\!]} \cong F(o)$. We define $\psi_o = \alpha_F^{-1} : \overline{[\![o]\!]} \cong F(o)$ in $\mathrm{Ty}_\mathcal{C}(1)$.

Uniqueness of (φ, ψ) Let $(\varphi', \psi') : \overline{[\![\cdot]\!]} \to F$ be another pseudo cwf-transformation in $\mathbf{CwF}^o$. We prove the following by induction:

- If $\Gamma \vdash$, then $\varphi_\Gamma = \varphi'_\Gamma$
- If $\Gamma \vdash A$, then $\psi_A = \psi'_A$

[1] PD: we should add equivalence class brackets in many places
[2] Reviewer 2: explain notation

Empty context There is a unique morphism between the two terminal objects $\overline{[\![1]\!]}$ and $F1$, thus $\varphi_1 = \varphi'_1$.

Context extension Assume by induction $\varphi_\Gamma = \varphi'_\Gamma$ and $\psi_A = \psi'_A$. By the coherence law of pseudo cwf-transformations, we have $\varphi'_{\Gamma.A} = \rho^{-1} \circ {\varphi'_\Gamma}^+ \circ \psi'_A$ from which the equality $\varphi'_{\Gamma.A} = \varphi_{\Gamma.A}$ follows.

Type substitution Assume we have $\Delta \vdash A$ and $\Gamma \vdash \gamma : \Delta$, and consider $\psi_{A\gamma}$ and $\psi'_{A\gamma}$. By definition of pseudo cwf-transformations, one has $\mathbf{T}(\varphi'_\Gamma)(\theta^{-1}_{A,\gamma}) \circ \psi'_{A\gamma} = \mathbf{T}(F\gamma)(\psi'_A)$. Since we know $\varphi_\Gamma = \varphi'_\Gamma$ we know φ'_Γ is an isomorphism and thus $\psi'_{A\gamma}$ depends only on φ'_Γ and ψ'_A from which it follows that $\psi'_{A\gamma} = \psi'_{A\gamma}$.

Base type The definition of 2-cells in $\mathbf{CwF}^o$ (as $\overline{[\![o]\!]} = o_\mathcal{C}$) entails $\psi'_o = \alpha_F^{-1} : \overline{[\![o]\!]} \to F([\![o]\!])$.

4 A free lccc

4.1 From cwfs to lcccs

We now extend our cwf-calculus with extensional I-types, N_1, Σ, and Π and prove that it yields a free cwf supporting these type formers. In order to show that this yields a free lccc we apply the biequivalence [5] between lcccs and *democratic* cwfs supporting these type formers.

▶ **Definition 13** (Democratic cwfs)**.** A cwf $\mathcal{C}$ is democratic when for each context Γ there is a type $\bar{\Gamma} \in \mathrm{Ty}(1)$ such that $\Gamma \cong 1.\bar{\Gamma}$. A pseudo cwf morphism $F : \mathcal{C} \to \mathcal{D}$ between democratic cwf preserves democracy when there is an isomorphism $F(\bar{\Gamma}) \cong \overline{F\Gamma}\langle\rangle_\Gamma$ satisfying a coherence diagram stated in [5] (Definition 8).

The free cwf with N_1, Σ, and Π-types is democratic since the empty context can be represented by the unit type N_1 and context extension by a Σ-type.

4.2 Cwfs with support for type constructors

A cwf supports a certain type former if it has extra structure and equations corresponding to the formation, introduction, elimination, and equality rules for the type former in question. We only spell out what it means for a cwf to support extensional identity types and refer the reader to [7, 5] for the definitions of what it means for a cwf to support and preserve Σ- and Π-types. The definition of what it means to support and preserve N_1 is analogous.

▶ **Definition 14** (Cwf with identity types)**.** A cwf $\mathcal{C}$ is said to support extensional identity types when for each $a, a' \in \mathrm{Tm}_\mathcal{C}(\Gamma, A)$ there is a type $\mathrm{I}(A, a, a')$ satisfying the following condition:

- $\mathrm{I}(A, a, a')[\gamma] = \mathrm{I}(A[\gamma], a[\gamma], a'[\gamma])$ for any $\gamma : \Delta \to \Gamma$
- For $a \in \mathrm{Tm}_\mathcal{C}(\Gamma, A)$, there is $\mathrm{r}(a) \in \mathrm{Tm}_\mathcal{C}(\Gamma, \mathrm{I}(A, a, a))$. Moreover, if $c \in \mathrm{Tm}_\mathcal{C}(\Gamma, \mathrm{I}(A, a, a'))$ then $a = a'$ and $c = \mathrm{r}(a)$.

A pseudo cwf morphism F preserves identity types when $\mathrm{I}(FA, Fa, Fa') \cong_\Gamma F(\mathrm{I}(A, a, a'))$.

We write $\mathbf{CwF}_d^{\Sigma,\Pi,I}$ for the 2-category of democratic cwfs supporting Σ, Π and identity types with morphisms preserving them, and $\mathbf{CwF}_{s,d}^{\Sigma,\Pi,I}$ for the strict version. Note that by democracy, any democratic cwf has a unit type representing the empty context.

Σ and Π are functorial and preserve isomorphisms:

▶ **Lemma 15** (Functoriality of Σ). *Let $f_A : A \cong A'$ over a context $\Gamma \in \mathcal{C}$ and $f_B : B \cong B'[f_A]$ over $\Gamma.A$, with $\Gamma.A \vdash B$ and $\Gamma.A' \vdash B'$. Then there exists a type isomorphism $\Sigma(A, B) \cong \Sigma(A', B')$ which is functorial meaning that if we have $g_A : A' \cong A''$ and $g_B : B' \cong B''[g_A]$, then $\Sigma(g_A \circ f_A, g_B \circ f_A^+ \circ f_B) = \Sigma(g_A, g_B) \circ \Sigma(f_A, f_B) : \Sigma(A, B) \to \Sigma(A'', B'')$.*

▶ **Lemma 16** (Functoriality of Π). *Let $A, A' \in \mathrm{Ty}_{\mathcal{C}}(\Gamma)$, $B \in \mathrm{Ty}_{\mathcal{C}}(\Gamma.A)$ and $B' \in \mathrm{Ty}_{\mathcal{C}}(\Gamma.A')$. Assume morever a type isomorphism $f_A : A \cong A'$ in $\boldsymbol{T}\Gamma$ and $f_B : B \cong B'[f_A]$ in $\boldsymbol{T}(\Gamma.A)$. Then there is a type isomorphism $\Pi(f_A, f_B) : \Pi(A, B) \cong \Pi(A', B')$ in $\boldsymbol{T}\Gamma$ such that for any term $\Gamma \vdash t : \Pi(A, B)$:*

$$\{\Pi(f_A, f_B)\}(t) = \lambda((\{f_B\}\mathrm{ap}(t))[f_A^{-1}])$$

and functorial in the same sense as for Σ-types.

4.3 The syntactic cwf with extensional I, N_1, Σ, and Π.

We extend the grammar and the set of inference rules with rules for $\mathrm{I}, \mathrm{N}_1, \Sigma$, and Π-types:

$$\begin{array}{lll} A & ::= & \cdots \mid \mathrm{I}(a, a) \mid \mathrm{N}_1 \mid \Sigma(A, A) \mid \Pi(A, A) \\ a & ::= & \cdots \mid \mathrm{r}(a) \mid 0_1 \mid \mathrm{fst}(A, a) \mid \mathrm{snd}(A, A, a) \mid \mathrm{pair}(A, A, a, a) \mid \mathrm{ap}(A, A, a, a) \mid \lambda(A, a) \end{array}$$

For each type we define its context:

$$\mathrm{ctx\text{-}of}(\mathrm{I}(a, a')) = \mathrm{ctx\text{-}of}(\mathrm{type\text{-}of}(a)) \qquad \mathrm{ctx\text{-}of}(\Sigma(A, B)) = \mathrm{ctx\text{-}of}(A)$$
$$\mathrm{ctx\text{-}of}(\Pi(A, B)) = \mathrm{ctx\text{-}of}(A)$$

For each term we define its type:

$$\mathrm{type\text{-}of}(\mathrm{fst}(A, c)) = A \qquad \mathrm{type\text{-}of}(\mathrm{r}(a)) = \mathrm{I}(a, a)$$
$$\mathrm{type\text{-}of}(\mathrm{snd}(A, B, c) = B\,\langle \mathtt{id}_{\mathrm{ctx\text{-}of}(A)}, \mathrm{fst}(A, c)\rangle \qquad \mathrm{type\text{-}of}(\lambda(A, c)) = \Pi(A, \mathrm{type\text{-}of}(c))$$
$$\mathrm{type\text{-}of}(\mathrm{pair}(A, B, a, b)) = \Sigma(A, B) \qquad \mathrm{type\text{-}of}(\mathrm{ap}(A, B, c, a)) = B\,\langle \mathtt{id}_{\mathrm{ctx\text{-}of}(A)}, a\rangle$$

4.3.1 Inference rules.

Rules for I-types:

$$\frac{\Gamma \vdash a = a' : A \qquad \Gamma \vdash b = b' : A}{\Gamma \vdash \mathrm{I}(a, b) = \mathrm{I}(a', b')} \qquad \frac{\Gamma \vdash a = a' : A}{\Gamma \vdash \mathrm{r}(a) = \mathrm{r}(a') : \mathrm{I}(a, a')} \qquad \frac{\Gamma \vdash c : \mathrm{I}(a, a')}{\Gamma \vdash a = a' : \mathrm{type\text{-}of}(a)}$$

$$\frac{\Gamma \vdash c : \mathrm{I}(a, a')}{\Gamma \vdash c = \mathrm{r}(a) : \mathrm{I}(a, a')} \qquad \frac{\Gamma \vdash a : A \qquad \Gamma \vdash a' : A \qquad \Delta \vdash \gamma : \Gamma}{\Gamma \vdash \mathrm{I}(a, a')\,\gamma = \mathrm{I}(a\,\gamma, a'\,\gamma)}$$

Rules for N_1:

$$\frac{}{\vdash \mathrm{N}_1} \qquad \frac{}{\vdash 0_1 : \mathrm{N}_1} \qquad \frac{\vdash a : \mathrm{N}_1}{\vdash a = 0_1 : \mathrm{N}_1}$$

Rules for Σ-types:

$$\frac{\Gamma \vdash A = A' \quad \Gamma.A \vdash B = B'}{\Gamma \vdash \Sigma(A,B) = \Sigma(A',B')} \qquad \frac{\Gamma \vdash A = A' \quad \Gamma \vdash c = c' : \Sigma(A,B)}{\Gamma \vdash \mathrm{fst}(A,c) = \mathrm{fst}(A',c') : A}$$

$$\frac{\Gamma \vdash A = A' \quad \Gamma.A \vdash B = B' \quad \Gamma \vdash c = c' : \Sigma(A,B)}{\Gamma \vdash \mathrm{snd}(A,B,c) = \mathrm{snd}(A',B',c') : B\,\langle \mathtt{id}_{\text{ctx-of}(A)}, \mathrm{fst}(A,c)\rangle}$$

$$\frac{\Gamma \vdash A = A' \quad \Gamma.A \vdash B = B' \quad \Gamma \vdash a = a' : A' \quad \Gamma \vdash b = b' : B\,\langle \mathtt{id}_{\text{ctx-of}(A)}, \mathrm{fst}(A,c)\rangle}{\Gamma \vdash \mathrm{pair}(A,B,a,b) = \mathrm{pair}(A',B',a',b') : \Sigma(A,B)}$$

$$\frac{\Gamma \vdash A \quad \Gamma.A \vdash B \quad \Gamma \vdash a : A' \quad \Gamma \vdash b : B\,\langle \mathtt{id}_{\text{ctx-of}(A)}, \mathrm{fst}(A,c)\rangle}{\Gamma : \mathrm{fst}(A, \mathrm{pair}(A,B,a,b)) = a : A}$$

$$\frac{\Gamma \vdash A \quad \Gamma.A \vdash B \quad \Gamma \vdash a : A' \quad \Gamma \vdash b : B\,\langle \mathtt{id}_{\text{ctx-of}(A)}, \mathrm{fst}(A,c)\rangle}{\Gamma : \mathrm{snd}(A,B, \mathrm{pair}(A,B,a,b)) = b : A}$$

$$\frac{\Gamma \vdash A \quad \Gamma.A \vdash B \quad \Gamma \vdash a : A \quad \Gamma \vdash b : B\,\langle \mathtt{id}_{\text{ctx-of}(A)}, \mathrm{fst}(A,c)\rangle}{\Gamma \vdash c = \mathrm{pair}(A,B,\mathrm{fst}(A,c),\mathrm{snd}(A,B,c)) : \Sigma(A,B)}$$

$$\frac{\Gamma \vdash A \quad \Gamma.A \vdash B \quad \Delta \vdash \gamma : \Gamma}{\Gamma \vdash \Sigma(A,B)\,\gamma = \Sigma(A\,\gamma, B\,\gamma^+)} \qquad \frac{\Gamma \vdash A \quad \Gamma \vdash c : \Sigma(A,B) \quad \Delta \vdash \gamma : \Gamma}{\Gamma \vdash \mathrm{fst}(A,c)\,\gamma = \mathrm{fst}(A\,\gamma, c\,\gamma) : A}$$

$$\frac{\Gamma \vdash A \quad \Gamma.A \vdash B \quad \Gamma \vdash c : \Sigma(A,B) \quad \Delta \vdash \gamma : \Gamma}{\Gamma \vdash \mathrm{snd}(A,B,c)\,\gamma = \mathrm{snd}(A\,\gamma, B\,\gamma^+, c\,\gamma) : B\langle \gamma, \mathrm{fst}(A,c)\gamma\rangle}$$

$$\frac{\Gamma \vdash A \quad \Gamma.A \vdash B \quad \Gamma \vdash a : A \quad \Gamma \vdash b : B\,\langle \mathtt{id}_{\text{ctx-of}(A)}, \mathrm{fst}(A,c)\rangle \quad \Delta \vdash \gamma : \Gamma}{\Gamma \vdash \mathrm{pair}(A,B,a,b)\,\gamma = \mathrm{pair}(A\,\gamma, B\,\gamma^+, a\,\gamma, b\,\gamma) : \Sigma(A,B)\gamma}$$

Rules for Π-types.

$$\frac{\Gamma \vdash A = A' \quad \Gamma.A \vdash B = B'}{\Gamma \vdash \Pi(A,B) = \Pi(A',B')} \qquad \frac{\Gamma \vdash A = A' \quad \Gamma.A \vdash t = t' : B}{\Gamma \vdash \lambda(A,t) = \lambda(A',t') : \Pi(A,B)}$$

$$\frac{\Gamma \vdash A = A' \quad \Gamma.A \vdash B = B' \quad \Gamma \vdash t = t' : \Pi(A,B) \quad \Gamma \vdash u = u' : A}{\Gamma \vdash \mathrm{app}(A,B,t,u) = \mathrm{app}(A',B',t',u') : B\langle \mathtt{id}, u\rangle}$$

$$\frac{\Gamma \vdash t : \Pi(A,B) \quad \Gamma \vdash u : A}{\Gamma \vdash \mathrm{app}(A,B,\lambda(A,t),u) = t\langle \mathtt{id}, u\rangle : B\langle \mathtt{id}, u\rangle} \qquad \frac{\Gamma \vdash t : \Pi(A,B)}{\Gamma \vdash \lambda(\mathrm{app}(t[p],q)) = t : \Pi(A,B)}$$

$$\frac{\Gamma \vdash A \quad \Gamma.A \vdash B \quad \Delta \vdash \gamma : \Gamma}{\Delta \vdash \Pi(A,B)\gamma = \Pi(A\gamma, A\gamma^+)} \qquad \frac{\Gamma \vdash t : \Pi(A,B) \quad \Delta \vdash \gamma : \Gamma}{\Delta \vdash \lambda(t)\gamma = \lambda(t\gamma^+) : \Pi(A,B)\gamma}$$

$$\frac{\Gamma \vdash t : \Pi(A,B) \quad \Gamma \vdash u : A \quad \Delta \vdash \gamma : \Gamma}{\Delta \vdash \mathrm{app}(t,u)\gamma = \mathrm{app}(t\gamma, u\gamma) : B\langle \gamma, u\gamma\rangle}$$

We remark that some of the inference rules can be derived from others.

4.3.2 The syntactic cwf supporting $\mathrm{I}, \mathrm{N}_1, \Sigma,$ and Π

It is straightforward to extend the definition of the term model $\mathcal{T}$ with $\mathrm{N}_1, \Sigma, \Pi$ and identity types, to form a cwf $\mathcal{T}^{\mathrm{I},\mathrm{N}_1,\Sigma,\Pi}$ supporting these type constructors. As we already explained it is democratic.

We want to show that $\mathcal{T}^{\mathrm{I},\mathrm{N}_1,\Sigma,\Pi}$ is free, not only in the 2-category of cwfs supporting these type formers, but in the subcategory of the democratic ones. It is straightforward to extend the interpretation functor and prove its uniqueness. It is also easy to check that it preserves democracy.

▶ **Theorem 17.** $\mathcal{T}^{\mathrm{I},\mathrm{N}_1,\Sigma,\Pi}$ *is the free democratic cwf supporting* $\mathrm{I},\mathrm{N}_1,\Sigma,\Pi$ *on one object.*

4.4 Bifreeness of $\mathcal{T}^{\Sigma,\Pi,\mathrm{N}_1,\mathrm{I}}$

We now prove the key result: that $\mathcal{T}^{\Sigma,\Pi,\mathrm{N}_1,\mathrm{I}}$ is the bifree cwf on one object in the 2-category $\mathbf{CwF}_d^{\Sigma,\Pi,I}$. This will let us prove that is the category of contexts of the cwf is a free lccc on one object in the 2-category of lccc.

▶ **Theorem 18.** *The cwf* $\mathcal{T}^{\Sigma,\Pi,\mathrm{N}_1,\mathrm{I}}$ *is bi-initial in the 2-category* $\boldsymbol{CwF}_d^{\Sigma,\Pi,I,o}$ *built similarly as* $\boldsymbol{CwF}^o$ *(see Definition 11).*

The proof goes as for the cwf case, and we only extend the induction.

4.4.1 Existence.

We resume our inductive proof from Section 3.4 with the cases for Π, Σ, N_1, and identity types.

Unit type F preserving democracy entails $1.F(N_1) = 1.F(\bar{1}) \cong \overline{F1} \cong \bar{1}$ (F preserves terminal objects).

Identity type Assume $\Gamma \vdash a, a' : A$. By induction, we have $\psi_A : \overline{[\![A]\!]} \cong FA[\varphi_\Gamma]$ in the indexed category over $\overline{[\![\Gamma]\!]}$. We know identity types preserve isomorphisms in the indexed category (Lemma 10, page 35 of [5]) yielding

$$\psi_{\mathrm{I}(A,a,a')} : \overline{[\![\mathrm{I}(A,a,a')]\!]} \cong \mathrm{I}_{\mathcal{C}}(FA[\varphi_\Gamma], \{\psi_A\}(\overline{[\![a]\!]}), \{\psi_A\}(\overline{[\![a']\!]})) = \mathrm{I}_{\mathcal{C}}(FA[\varphi_\Gamma], F(a)[\varphi_\Gamma], F(a')[\varphi_\Gamma])$$

.

Σ-types Assume we have $\Gamma \vdash A$ and $\Gamma \cdot A \vdash B$. By induction we have the isomorphism $\psi_A : \overline{[\![A]\!]} \cong_\Gamma FA[\varphi_\Gamma]$ and $\psi_B : \overline{[\![B]\!]} \cong_{\Gamma.A} FB[\varphi_{\Gamma.B}]$.
We let

$$\psi_{\Sigma(A,B)} = \Gamma.\Sigma(A,B) \xrightarrow{\Sigma(\psi_A,\psi_B)} \Gamma.\Sigma(FA[\varphi_\Gamma], FB[\rho^{-1} \circ \varphi_\Gamma^{-1^+}]) \xrightarrow{\mathbf{T}(\varphi_\Gamma)(s_{A,B}^{-1})} \Gamma.F(\Sigma(A,B))[\varphi_\Gamma]$$

$\psi_{\Sigma(A,B)}$ can be related to $\varphi_{\Gamma.A.B}$:

$$\begin{aligned}
\psi_{\Sigma(A,B)} &= \mathbf{T}(\varphi_\Gamma)(s_{A,B}^{-1}) \circ \Sigma(\psi_A, \psi_B)\\
&= \varphi_\Gamma^{-1^+} \circ s_{A,B}^{-1} \circ \varphi_\Gamma{}^+ \circ \chi^{-1} \circ \psi_A^+ \circ \psi_B \circ \chi_{A,B}\\
&= \varphi_\Gamma^{-1^+} \circ s_{A,B}^{-1} \circ \chi_{A,B} \circ \varphi_\Gamma{}^{++} \circ \psi_A^+ \circ \psi_B \circ \chi_{A,B}\\
&= \varphi_\Gamma^{-1^+} \circ \rho \circ F(\chi_{A,B}) \circ \rho^{-1} \circ \rho^{-1^+} \circ \varphi_\Gamma{}^{++} \circ \psi_A^+ \circ \psi_B \circ \chi_{A,B}\\
&= \varphi_\Gamma^{-1^+} \circ \rho \circ F(\chi_{A,B}^{-1}) \circ \rho^{-1} \circ \varphi_{\Gamma.A}^+ \circ \psi_B \circ \chi_{A,B}\\
&= \varphi_\Gamma^{-1^+} \circ \rho \circ F(\chi_{A,B}^{-1}) \circ \varphi_{\Gamma.A.B} \circ \chi_{A,B}
\end{aligned}$$

From that calculation, we deduce

$$\varphi_{\Gamma.\Sigma(A,B)} = F(\chi_{A,B}^{-1}) \circ \varphi_{\Gamma.A.B} \circ \chi_{A,B}$$

Π-types Define $\psi_{\Pi(A,B)}$ as follows

$$\begin{aligned}\overline{[\![\Gamma.\Pi(A,B)]\!]} &\xrightarrow{\Pi(\psi_A,\psi_B)} \overline{[\![\Gamma]\!]}.\Pi(FA[\varphi_\Gamma], FB[\rho\circ\varphi_{\Gamma\uparrow FA}]) \\ &= \overline{[\![\Gamma]\!]}.\Pi(FA, FB[\rho])[\varphi_\Gamma] \\ &\xrightarrow{T(\varphi_\Gamma)(\xi^{-1}_{A,B})} \overline{[\![\Gamma]\!]}.F(\Pi(A,B))[\varphi_\Gamma]\end{aligned}$$

4.4.2 Uniqueness

We resume the uniqueness proof left at 3.4.

Unit type Follows from the coherence diagram of the preservation of democracy of F.

Identity types We need to show $\psi'_{\mathrm{I}(A,a,a')} = \psi_{\mathrm{I}(A,a,a')} : \Gamma.\mathrm{I}(A,a,a') \to \Gamma.F(\mathrm{I}(A,a,a'))[\varphi_\Gamma]$. By post-composing by the coherence isomorphism $F(\mathrm{I}(A,a,a')) \cong_{F\Gamma} \mathrm{I}(FA, Fa, Fa')$, we get a morphism between identity types. In an extensional type theory, identity types are either empty or singletons, thus there is at most one morphism between two identity types (which is an isomorphism). This implies that $\psi_{\mathrm{I}(A,a,a')} = \psi'_{\mathrm{I}(A,a,a')}$.

Σ-types By induction, we can assume that $\varphi_{\Gamma.A.B} = \varphi'_{\Gamma.A.B}$. By naturality of φ', we must have $\varphi'_{\Sigma(A,B)} = F(\chi^{-1}_{A,B}) \circ \varphi'_{\Gamma.A.B} \circ \chi_{A,B} = \varphi_{\Gamma.\Sigma(A,B)}$ hence $\psi_{\Sigma(A,B)} = \psi'_{\Sigma(A,B)}$ follows.

Π-types As in the previous section, by induction we can assume $\varphi_{\Gamma.A.B} = \varphi'_{\Gamma.A.B}$. Write ev for the obvious map $\Gamma.A.\Pi(A,B)[p] \to \Gamma.A.B$ given by $\langle p, \mathrm{ap}(q)[\langle \mathtt{id}, q[p]\rangle]\rangle$. We have this lemma:

▶ **Lemma 19.** *Assume $\Gamma.A \vdash B$. The only automorphism ω of $\Pi(A,B)$ (in $\boldsymbol{T}\Gamma$) such that $Tp(\omega) : \Gamma.A.\Pi(A,B)p \cong \Gamma.A.\Pi(A,B)p$ is such that $ev \circ Tp(\omega) = ev$ is the identity.*

Proof. Consequence of Proposition 11 of [5]. ◀

To conclude from this lemma, we need only show that $\psi^{-1}_{\Pi(A,B)} \circ \psi'_{\Pi(A,B)}$ satisfies the condition. But we have:

$$\begin{aligned}F(ev)\circ\rho^{-1}\circ\theta_{\Pi(A,B),p}\circ\varphi_{\Gamma.A}\circ Tp(\psi'_{\Pi(A,B)}) &= F(ev)\circ\rho^{-1}\circ\varphi_{\Gamma.A}\circ T(\varphi_{\Gamma.A})(\theta)\circ Tp(\psi'_{\Pi(A,B)}) \\ &= F(ev)\circ\rho^{-1}\circ\varphi_{\Gamma.A}\circ\psi_{\Pi(A,B)[p]} \\ &= F(ev)\circ\varphi_{\Gamma.A.\Pi(A,B)p} \\ &= \varphi'_{\Gamma.A.B}\circ ev\end{aligned}$$

By only using naturality conditions on φ' and ψ'. Write $\tau : \Gamma.A.F(\Pi(A,B))[\varphi_\Gamma][p] \to F(\Gamma.A.B)$ for the map $F(ev)\circ\rho^{-1}\circ\theta_{\Pi(A,B),p}\circ\varphi_{\Gamma.A}$. Since φ and ψ are natural, we can do the same reasoning, and have $\tau\circ Tp(\psi_{\Pi(A,B)}) = \varphi_{\Gamma.A.B}\circ ev$. Thus, we get:

$$\varphi^{-1}_{\Gamma.A.B}\circ\tau = ev\circ Tp(\psi^{-1}_{\Pi(A,B)})$$

Using our induction hypothesis on B ($\varphi_{\Gamma.A.B} = \varphi'_{\Gamma.A.B}$) we have

$$ev\circ Tp(\psi^{-1}_{\Pi(A,B)})\circ Tp(\psi'_{\Pi(A,B)}) = \varphi^{-1}_{\Gamma.A.B}\circ\tau\circ Tp(\psi'_{\Pi(A,B)}) = ev$$

as desired, thus $\psi_{\Pi(A,B)} = \psi'_{\Pi(A,B)}$.

4.5 The free lccc

Let **LCC** be the 2-category of lcccs. Because of biequivalence [5] we have pseudofunctors:

$$U : \mathbf{CwF}_d^{\Sigma,\Pi,I} \to \mathbf{LCC} \quad H : \mathbf{LCC} \to \mathbf{CwF}_d^{\Sigma,\Pi,\mathrm{I}}$$

such that $UH = \mathrm{I}$ and $HU \cong \mathrm{I}$. In particular there are adjunctions $H \dashv U$ and $U \dashv H$.

▶ **Theorem 20.** *$U\mathcal{T}^{\Sigma,\Pi,\mathrm{N}_1,I}$ is the bifree lccc on one object, ie. it is bi-initial in $\boldsymbol{LCC^o}$.*

Proof. Let $\mathbb{C}$ be a **LCC** with a chosen object $o_{\mathbb{C}} \in \mathbb{C}$. By democracy, $o_{\mathbb{C}}$ can be seen as a type over the empty context in the cwf $H\mathbb{C}$, thus $(H\mathbb{C}, o)$ is in $\mathbf{CwF}_d^{\Sigma,\Pi,I,o}$. Thus we have a pseudo cwf functor $\overline{[\![\cdot]\!]} : \mathcal{T}^{\Sigma,\Pi,\mathrm{N}_1,I} \to H\mathbb{C}$ satisfying $[\![o]\!] \cong o_{\mathbb{C}}$. which through the adjunction yields $F : U\mathcal{T}^{\Sigma,\Pi,\mathrm{N}_1,I} \to \mathbb{C}$ in **LCC**.

Assume we have another $G : U\mathcal{T}^{\Sigma,\Pi,\mathrm{N}_1,\mathrm{I}} \to \mathbb{C}$, through the adjunction we get $G^* : \mathcal{T}^{\Sigma,\Pi,\mathrm{N}_1,\mathrm{I}} \to H\mathbb{C}$. Thus by bifreeness of $\mathcal{T}^{\Sigma,\Pi,\mathrm{N}_1,I}$ we have $\varphi : \overline{[\![\cdot]\!]} \cong G^*$, thus $F \cong G$. Moreover, any other morphism $F \to G$ yields a morphism $\overline{[\![\cdot]\!]} \to G^*$ and is equal to φ. ◀

References

1 Henk P. Barendregt. Lambda calculi with types. In Samson Abramsky, Dov Gabbay, and Tom Maibaum, editors, *Handbook of Logic in Computer Science*, volume 2, pages 118–310. Oxford University Press, 1992.

2 Jean Benabou. Fibered categories and the foundations of naive category theory. *J. Symb. Log*, 50(1):10–37, 1985.

3 John Cartmell. Generalized algebraic theories and contextual categories. *Annals of Pure and Applied Logic*, 32:209–243, 1986.

4 Pierre Clairambault and Peter Dybjer. The biequivalence of locally cartesian closed categories and martin-löf type theories. In *Typed Lambda Calculi and Applications - 10th International Conference, TLCA 2011, Novi Sad, Serbia, June 1-3, 2011. Proceedings*, pages 91–106, 2011.

5 Pierre Clairambault and Peter Dybjer. The biequivalence of locally cartesian closed categories and martin-löf type theories. *Mathematical Structures in Computer Science*, 24(6), 2014.

6 Pierre-Louis Curien. Substitution up to isomorphism. *Fundamenta Informaticae*, 19(1,2):51–86, 1993.

7 Peter Dybjer. Internal type theory. In *TYPES '95, Types for Proofs and Programs*, number 1158 in Lecture Notes in Computer Science, pages 120–134. Springer, 1996.

8 Martin Hofmann. Interpretation of type theory in locally cartesian closed categories. In *Proceedings of CSL*. Springer LNCS, 1994.

9 Per Martin-Löf. Constructive mathematics and computer programming. In *Logic, Methodology and Philosophy of Science, VI, 1979*, pages 153–175. North-Holland, 1982.

10 Per Martin-Löf. Substitution calculus. Notes from a lecture given in Göteborg, November 1992.

11 Erik Palmgren and Steve J. Vickers. Partial horn logic and cartesian categories. *Annals of Pure and Applied Logic*, 145(3):314 – 353, 2007.

12 Robert Seely. Locally cartesian closed categories and type theory. *Math. Proc. Cambridge Philos. Soc.*, 95(1):33–48, 1984.

13 Thomas Streicher. Semantics of type theory. In *Progress in Theoretical Computer Science*, number 12. Basel: Birkhaeuser Verlag, 1991.

14 Alvaro Tasistro. Formulation of Martin-Löf's theory of types with explicit substitutions. Technical report, Department of Computer Sciences, Chalmers University of Technology and University of Göteborg, 1993. Licentiate Thesis.

A Remark on the definition of pseudo cwf-morphisms (erratum for [5])

In [5], pseudo cwf-morphisms (2-cells in the 2-category of cwfs) are defined as follows

▶ **Definition 21** (Pseudo cwf-transformation)**.** Let F and G be two cwf-morphisms from $(\mathcal{C}, T)$ to $(\mathcal{C}', T')$. A *pseudo cwf-transformation* from F to G is a pair (ϕ, ψ) where $\phi : F \Rightarrow G$ is a natural transformation, and for each Γ in $\mathcal{C}$ and $A \in \mathrm{Ty}(\Gamma)$, a morphism $\psi_{\Gamma,A} : FA \to GA[\phi_\Gamma]$ in $\mathbf{T}'(F\Gamma)$, natural in A and such that the following diagram commutes:

$$\begin{array}{ccc}
FA[F\delta] & \xrightarrow{\mathbf{T}'(F\delta)(\psi_{\Gamma,A})} & GA[\phi_\Gamma F(\delta)] \\
\downarrow{\scriptstyle \theta_{A,\delta}} & & \downarrow{\scriptstyle \mathbf{T}'(\phi_\Delta)(\theta'_{A,\delta})} \\
F(A[\delta]) & \xrightarrow[\psi_{\Delta,A[\delta]}]{} & G(A[\delta])[\phi_\Delta]
\end{array}$$

where θ and θ' are the isomorphisms witnessing preservation of substitution in types in the definition of pseudo cwf-morphism.

In the process of developping the present contribution, we discovered a glitch with this definition: the component ψ is not constrained enough by ϕ. This causes a mismatch with the 2-cells in **LCC** (where only the ϕ remains), and in consequence the family of cwf-transformations ϵ used in the biequivalence (see [5]) fails a condition of pseudonatural transformations.

Missing from this definition is the following coherence diagram:

$$\begin{array}{ccccc}
F(\Gamma.A) & & \xrightarrow{\phi_{\Gamma.A}} & & G(\Gamma.A) \\
\downarrow{\scriptstyle \rho^F_{\Gamma,A}} & & & & \downarrow{\scriptstyle \rho^G_{\Gamma,A}} \\
F\Gamma.FA & \xrightarrow{\psi_{\Gamma,A}} & F\Gamma.FA[\phi_\Gamma] & \xrightarrow{\langle \phi_\Gamma \mathrm{p}, \mathrm{q} \rangle} & G\Gamma.GA
\end{array}$$

This means that ψ becomes redundant, and can be defined from ϕ – one could get rid of ψ and adopt natural transformations $\phi : F \Rightarrow G$ as 2-cells from F to G. We refrain from doing that because pseudo cwf-morphisms is most naturally presented with the ψ, reflecting the second components of cwfs and cwf-morphisms. Moreover in our proof of bifreeness, the construction of the unique cwf-transformation between the interpretation and an arbitrary pseudo cwf-functor naturally build ϕ in mutual induction with ψ.

Finally, we finish this erratum with two remarks:

(1) The pseudofunctor H of [5] yields cwf-transformations satisfying this diagram: in fact they are defined in this way.

(2) The coherence diagram in the original definition above follows from this, as is established in a straightforward adaptation of Lemma 5 in [5] (the proof uses the fact that G preserves finite limits which might not be the case in general, but as a pseudo cwf-morphism it always preserves the substitution pullback used in the proof).

B Proof of the bi-initiality of $\mathbf{CwF}_s$

In this section we complete the proof given in Section 3.4. To simplify notation, we will identity a syntactic term with its interpretation in $\mathcal{C}$ through $\overline{[\![\cdot]\!]}$.

Projection Assume we have $p : \Gamma.A \to \Gamma$. Then we need to check that $\varphi_{\Gamma.A}; Fp = p; \varphi_\Gamma$. This is a simple calculation:

$$\begin{aligned}
\varphi_{\Gamma.A}; Fp &= \psi_A; \langle \varphi_\Gamma, q\rangle; \rho^{-1}_{\Gamma,A}; Fp && \text{def. of } \varphi_{\Gamma.A}\\
&= \psi_A; \langle (p; \varphi_\Gamma), q\rangle; p && \text{properties of } \rho_{\Gamma,A}\\
&= \psi_A; p; \varphi_\Gamma = p; \varphi_\Gamma && \text{because } \psi_A \text{ is a map in } \mathbf{T}\Gamma
\end{aligned}$$

Extension Assume we have a $\Gamma \vdash f : \Delta$ and a $\Gamma \vdash t : A[f]$ so that $\langle f, t\rangle$ is a morphism from Γ to $\Delta \cdot A$.

$$\begin{aligned}
F\langle f, t\rangle \circ \varphi_\Gamma &= \rho^{-1} \circ \langle Ff, \{\theta_A^{-1}\}(Ft)\rangle \circ \varphi_\Gamma\\
&= \rho^{-1} \circ \langle Ff \circ \varphi_\Gamma, \mathbf{T}\varphi_\Gamma\{\theta_A^{-1}\}(Ft[\varphi_\Gamma])\rangle && \text{property of } \mathbf{T} \text{ wrt substitution}\\
&= \rho^{-1} \circ \langle \varphi_\Delta \circ f, \{\mathbf{T}\varphi_\Gamma(\theta_A^{-1})\}(\psi_{Af}(t))\rangle && \text{induction hypothesis on } f \text{ and } t\\
&= \rho^{-1} \circ \langle \varphi_\Delta \circ f, \mathbf{T}(f)(\psi_A)(t)\rangle && \text{def. of } \psi_{Af}\\
&= \rho^{-1} \circ \langle \varphi_\Delta \circ f, q[\psi_A \circ \langle f, t\rangle]\rangle && \text{def. of } \mathbf{T}\\
&= \rho^{-1} \circ \langle \varphi_\Delta \circ p, q\rangle \circ \psi_A \circ \langle f, t\rangle\\
&= \rho^{-1} \circ \varphi_\Delta \uparrow \circ \psi_A \circ \langle f, t\rangle = \varphi_{\Delta.A} \circ \langle f, t\rangle
\end{aligned}$$

Term substitution Assume we have $\Gamma \vdash f : \Delta$ and $\Delta \vdash t : A$. First, we have

$$\begin{aligned}
\{\mathbf{T}(\overline{[\![f]\!]})(\psi_A)\}(\overline{[\![tf]\!]}) &= q[\psi_A \circ (\overline{[\![f]\!]} \uparrow \overline{[\![A]\!]}) \circ \langle \mathtt{id}, \overline{[\![tf]\!]}\rangle]\\
&= q[\psi_A \circ \langle \mathtt{id}, \overline{[\![t]\!]}\rangle \circ f]\\
&= q[\psi_A \circ \langle \mathtt{id}, \overline{[\![t]\!]}\rangle][\overline{[\![f]\!]}]\\
&= \{\psi_A\}(\overline{[\![t]\!]})[\overline{[\![f]\!]}]\\
&= Ft[\varphi_\Delta][\overline{[\![f]\!]}] && \text{induction hypothesis on } t\\
&= Ft[Ff][\varphi_\Gamma] && \text{induction hypothesis on } f
\end{aligned}$$

Thus:

$$\begin{aligned}
\{\psi_{Af}\}(\overline{[\![tf]\!]}) &= \{\mathbf{T}(\varphi_\Gamma)(\theta_{A,f})(Ft[Ff][\varphi_\Gamma])\\
&= \theta_{A,f}(Ft[Ff])[\varphi_\Gamma] = F(t[f])[\varphi_\Gamma]
\end{aligned}$$

Variable Assume we have $\Gamma \cdot A \vdash q : A[p]$. Using the formula for ψ_{Ap} yields:

$$\begin{aligned}
\{\psi_{Af}\}(q) &= q[\theta_{A,p} \circ \langle \varphi_\Gamma, q[\psi_A \circ \langle p, q\rangle]\rangle]\\
&= q[\theta_{A,p} \circ \langle \varphi_{\Gamma \cdot A}, q[\psi_A]\rangle && \text{(because } \langle p, q\rangle = \mathtt{id}\text{')}
\end{aligned}$$

Calculating the other term yields:

$$\begin{aligned}
Fq[\varphi_{\Gamma \cdot A}] &= \{\theta_{A,p}\}(q[\rho])[\varphi_{\Gamma.A}] && \text{because } F \text{ preserves context comprehension}\\
&= q[\theta_{A,p} \circ \langle id, q[\rho]\rangle \circ \varphi_{\Gamma.A}]\\
&= q[\theta_{A,p} \circ \langle \varphi_{\Gamma.A}, q[\rho \circ \varphi_{\Gamma \cdot A}]\rangle]\\
&= q[\theta_{A,p} \circ \langle \varphi_{\Gamma.A}, q[\langle \varphi_\Gamma, q\rangle \circ \psi_A]\rangle] && \text{def. of } \varphi_{\Gamma.A}\\
&= q[\theta_{A,p} \circ \langle \varphi_{\Gamma.A}, q[\psi_A]\rangle]
\end{aligned}$$

Thus the equality holds.

Functoriality of substitution Assume we have $\Gamma \vdash f : \Delta$ and $\Delta \vdash g : \Theta$. We want to show the equality $\psi_{A[g][f]} = \psi_{A[f \circ g]}$ and $\psi_{A[id]} = \psi_A$ for $\Theta \vdash A$. The second equation is easy:

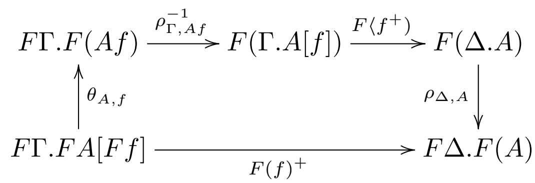
by functoriality of $\mathbf{T}$ $\mathbf{T}(\overline{[\![\mathtt{id}]\!]})(\psi_A) = \psi_A$ and properties of F, $\theta_{A,\mathtt{id}} = \mathtt{id}$. A calculation gives

$$\begin{aligned}
\psi_{A[g][f]} &= \mathbf{T}(f)(\mathbf{T}(g)(\psi_A) \circ T(\varphi_\Delta)(\theta_{A,g})) \circ \mathbf{T}(\varphi_\Gamma)(\theta_{A,f}) && \\
&= \mathbf{T}(f)(\mathbf{T}(g)(\psi_A)) \circ \mathbf{T}(f)(T(\varphi_\Delta)(\theta_{A,g})) \circ \mathbf{T}(\varphi_\Gamma)(\theta_{A,f} && \text{functoriality of } \mathbf{T}(f) \\
&= \mathbf{T}(g \circ f)(\psi_A) \circ \mathbf{T}(\varphi_\Gamma)(\mathbf{T}(Ff)(\theta_{A,g})) \circ \mathbf{T}(\varphi_\Gamma)(\theta_{A,f} && \text{functoriality of } \mathbf{T}(f) + \text{ind. hyp. on } f \\
&= \mathbf{T}(g \circ f)(\psi_A) \circ \mathbf{T}(\varphi_\Gamma)(\mathbf{T}(Ff)(\theta_{A,g}) \circ (\theta_{A,f})) && \text{functoriality of } \mathbf{T}(\varphi_\Gamma) \\
&= \mathbf{T}(g \circ f)(\psi_A) \circ \mathbf{T}(\varphi_\Gamma)(\theta_{A,g\circ f}) && \text{coherence for } \theta \\
&= \psi_{A[g \circ f]}
\end{aligned}$$

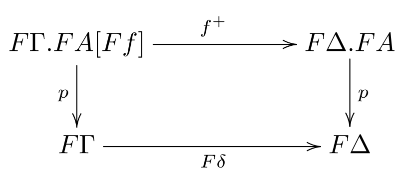
C Proof of the bi-initiality of $\mathcal{T}^{\Sigma,\Pi,I}$

In this section, we deal with the cases related to terms and equations arising for Σ, Π and identity types. For the equations dealing with substitutions and type constructor, we will need lemmas about the isomorphism of preservation of substitution of F, the type isomorphism $\theta_{A,f} : FA[Ff] \cong F(Af)$ (over $\mathbf{T}(F\Gamma)$). Those lemmas come from this characterization of θ:

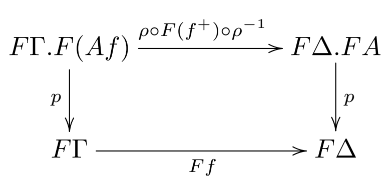
▶ **Lemma 22.** *Let $f : \Gamma \to \Delta$. The type morphism $\theta_{A,f}$ is the only type morphism to make the following diagram commute:*



Proof. The diagram commutes by virtue of Proposition 4 of [5]. Moreover, by definition of type substitution the following is pullback:



Because θ and ρ are isomorphism and the diagram above commute, the following is also a pullback:



Thus it follows that there is a unique type morphism $F\Delta.FA[Ff] \to F\Delta.F(Af)$ that makes the diagram of the lemma commute by the universal property of pullbacks. ◀

▶ **Lemma 23** (Compatibility of Σ-types with substitution). *The following diagram of type isomorphism over $F\Gamma$ commutes:*

$$\begin{array}{ccc} F(\Sigma(A,B))[Ff] & \xrightarrow{\theta_{\Sigma(A,B)}} & F(\Sigma(A,B)[f]) \\ \big\downarrow{\scriptstyle T(Ff)(s_{A,B})} & & \big\downarrow{\scriptstyle (s_{Af,B(f\uparrow\_)})} \\ \Sigma(FA[Ff],FB[\rho^{-1}])[Ff] & \xrightarrow[\Sigma(\theta_A,\boldsymbol{T}(\bar{\theta}_{A,f}^{-1})(\theta_B))]{} & \Sigma(F(Af),F(Bf\uparrow\_)) \end{array}$$

(It is well-typed because $\rho^{-1}\circ Ff\uparrow\_ = F(f\uparrow\_)\circ\rho^{-1}\circ\theta_A^{-1}$)

Proof. The diagram amounts to showing that $\theta_{\Sigma(A,B),f} = s_{Af,Bf^+}\circ\Sigma(\theta_A,T(\bar{\theta}_{A,f}))(\theta_B)\circ T(Ff)(s_{A,B})$. Hence by the previous lemma it is enough to show that the right hand side makes the corresponding diagram commute – which is an involved calculation. ◀

▶ **Lemma 24** (Compatibility of Σ with $\mathbf{T}$). *Assume we have $f_A : \Delta.A \cong \Delta.A'$ and $f_B : \Delta.A.B \cong \Delta.A.B'[f_A]$, and $f:\Gamma\to\Delta$. Then*

$$\boldsymbol{T}f(\Sigma(f_A,f_B)) = \Sigma(\boldsymbol{T}f(f_A),\boldsymbol{T}(f^+)(f_B)) : \Sigma(Af,Bf^+)\cong\Sigma(A'f,B'f^+)$$

in $\boldsymbol{T}\Gamma$.

Proof. Direct calculation. ◀

▶ **Lemma 25** (Compatibility of Π-types with substitution). *The following diagram of type isomorphism over $F\Gamma$ commutes:*

$$\begin{array}{ccc} F(\Pi(A,B))[Ff] & \xrightarrow{\theta_{\Pi(A,B)}} & F(\Pi(A,B)[f]) \\ \big\downarrow{\scriptstyle T(Ff)(\xi_{A,B})} & & \big\downarrow{\scriptstyle (\xi_{Af,B(f\uparrow\_)})} \\ \Pi(FA[Ff],FB[\rho^{-1}])[Ff] & \xrightarrow[\Pi(\theta_A,\boldsymbol{T}(\rho^{-1}\circ\theta_A^{-1})(\theta_B))]{} & \Pi(F(Af),F(Bf\uparrow\_)) \end{array}$$

(It is well-typed because $\rho^{-1}\circ Ff\uparrow\_ = F(f\uparrow\_)\circ\rho^{-1}\circ\theta_A^{-1}$)

Proof. Similar method to prove. ◀

▶ **Lemma 26** (Compatibility of Σ with $\mathbf{T}$). *Assume we have $f_A : \Delta.A \cong \Delta.A'$ and $f_B : \Delta.A.B \cong \Delta.A.B'[f_A]$, and $f:\Gamma\to\Delta$. Then*

$$\boldsymbol{T}f(\Pi(f_A,f_B)) = \Pi(\boldsymbol{T}f(f_A),\boldsymbol{T}(f^+)(f_B)) : \Pi(Af,Bf^+)\cong\Pi(A'f,B'f^+)$$

in $\boldsymbol{T}\Gamma$.

Proof. Direct calculation. ◀

Reflexivity Assume $\Gamma \vdash a : A$. We need to check that

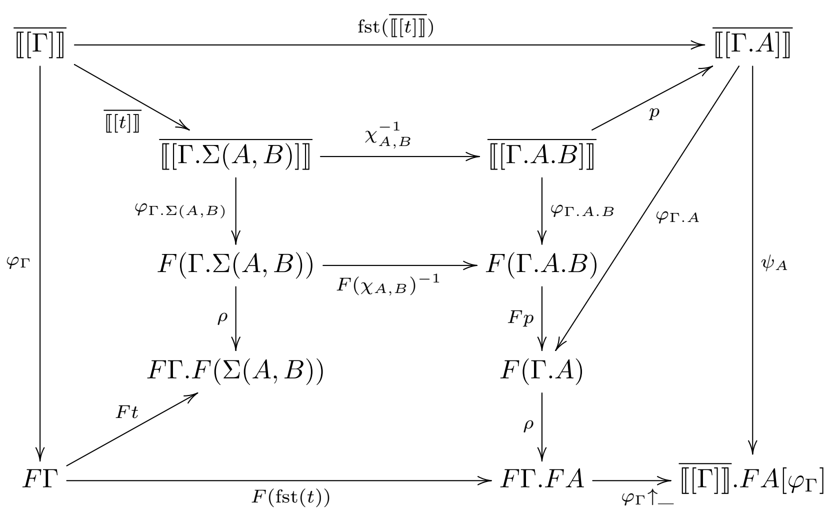
$$\{\psi_{\mathrm{I}(A,a,a)}\}(\overline{[\![\mathrm{r}(a)]\!]}) = F(\mathrm{r}(a))[\varphi_\Gamma]$$

But both are equal to $\mathrm{r}(\overline{[\![a]\!]})$ by the axiom of extension types.

Naturality of identity types Assume we have $\Delta \vdash a, a' : A$ and $\Gamma \vdash f : \Delta$. We need to show $\psi_{\mathrm{I}(A,a,a')[f]} = \psi_{\mathrm{I}(A[f],a[f],a'[f])}$. Both are isomorphisms between identity types but by extensionality there can be at most one isomorphism (mapping $\mathtt{r}(\overline{[\![a]\!]})$ to $\mathtt{r}(Fa)$) thus they have to be equal.

First projection Assume we have $\Gamma \vdash t : \Sigma(A, B)$ from which we deduce $\Gamma \vdash \mathrm{fst}(A, B, t) : A$.

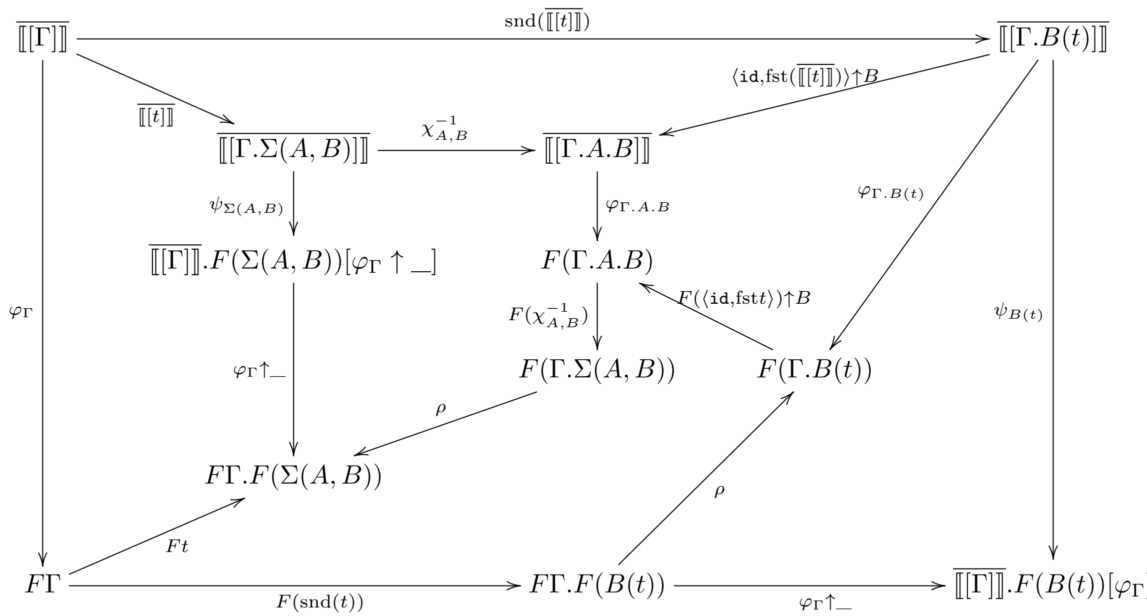
We check the diagram:



(we confused a term t and its section $\langle \mathtt{id}, t\rangle : \Gamma \to \Gamma.A$

Second projection Assume we have $\Gamma \vdash t : \Sigma(A, B)$ from which we deduce $\Gamma \vdash \mathrm{snd}(A, B, t)$:

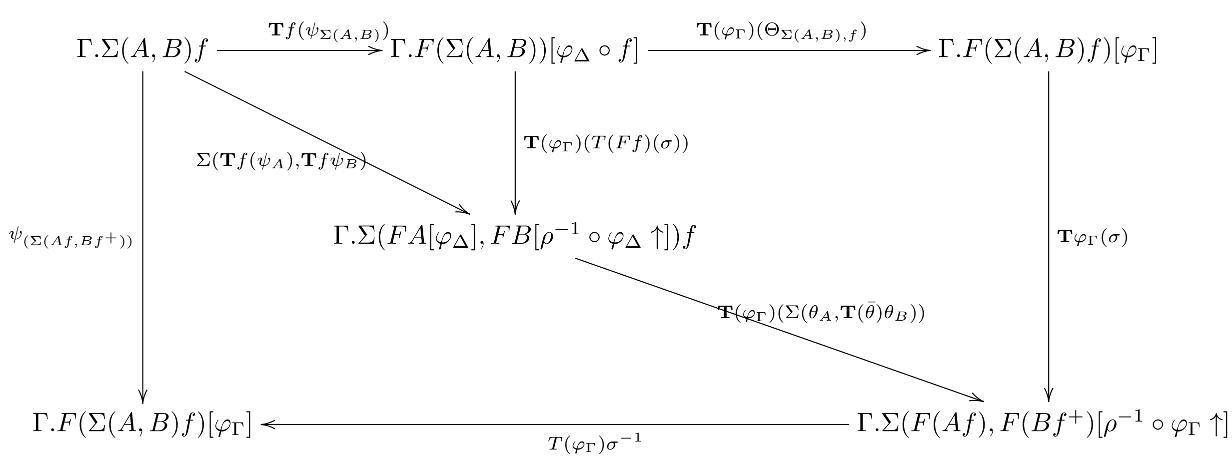
$B[\langle id, t\rangle]$. Writing $B(t)$ for $B\langle id, t\rangle$, we have the following diagram:



Left square is induction hypothesis on t, top square, bottom hexagon, and top-left square are basic cwf manipulations. Middle hexagon is the definition of $\psi_{\Sigma(A,B)}$, bottom right triangle is definition of $\varphi_{\Gamma.B(t)}$.

Naturality of the sigma types Assume we have $\Delta.A \vdash B$ and $\Gamma \vdash f : \Delta$. We want to prove equality of $\psi_{\Sigma(A,B)[f]}$ and $\psi_{\Sigma(Af,B(f\uparrow A))}$.

It follows from this diagram, where the top arrow is the unfolding of the definition of $\psi_{\Sigma(A,B)f}$.



The diagram commutes thanks to Lemmata 24, and 23.

λ Assume we have $\Gamma.A \vdash t : B$. Then we have

$$\begin{aligned}
\{\Pi(\psi_A,\psi_B)(\lambda(t)) &= \lambda(\{T(\psi_A^{-1})(\psi_B)\}(\mathrm{app}(\lambda(t[p\uparrow _]), q[\mathbf{T}(p)(\psi_A^{-1})]))) \\
&= \lambda(\{T(\psi_A^{-1})(\psi_B)\}(t[p\uparrow _ \circ \langle \mathtt{id}, q[\mathbf{T}(p)(\psi_A^{-1})])) \\
&= \lambda(\{T(\psi_A^{-1})(\psi_B)\}(t[\psi_A^{-1}])) \\
&= \lambda(q[\psi_B \circ \langle \psi_A^{-1}, t[\psi_A^{-1}]\rangle]) \\
&= \lambda(q[\psi_B \circ \langle \mathtt{id}, t\rangle \circ \psi_A^{-1}] \\
&= \lambda(\{\psi_B\}(t)[\psi_A^{-1}]) \\
&= \lambda(Ft[\varphi_{\Gamma.A} \circ \psi_A^{-1}]) = \lambda(Ft[\rho][\varphi_\Gamma \uparrow _]) = \lambda(Ft[\rho])[\varphi_\Gamma] \qquad \text{ind. hypothesis on } t
\end{aligned}$$

Thus, we get

$$\begin{aligned}
\{\psi_{\Pi(A,B)}\}(t) &= \{T(\varphi_\Gamma)(\xi_{A,B}^{-1})\}(\lambda(Ft(\rho))[\varphi_\Gamma] \\
&= q[\xi_{A,B}^{-1} \circ \langle \mathtt{id}, \lambda(Ft[\rho])\rangle \circ \varphi_\Gamma] \\
&= F(\lambda t)[\varphi_\Gamma] \qquad \text{because } F \text{ preserves } \Pi\text{-types}
\end{aligned}$$

ap Assume we have $\Gamma \vdash t : \Pi(A,B)$.

$$\begin{aligned}
F(\mathrm{ap}(t))[\varphi_{\Gamma.A}] &= \mathrm{ap}(\{\xi\}(Ft))[\varphi_\Gamma \uparrow _ \circ \psi_A] \\
&= \mathrm{ap}(\{\xi\}(Ft)[\varphi_\Gamma])[\psi_A] \\
&= \mathrm{ap}(\{\mathbf{T}(\varphi_\Gamma)(\xi)\}(\{\psi_{\Pi(A,B)}\}(t)))[\psi_A] \\
&= \mathrm{ap}(\{\Pi(\psi_A,\psi_B)\}(t))[\psi_A] \\
&= \{\psi_B\}\mathrm{ap}(t)
\end{aligned}$$

Naturality of Π-types Thanks to lemmata 26 and 25 we can draw the same diagram as for the naturality of Σ types but by replacing the structural morphisms of Σ-types by that of Π-types and the resulting diagram still commutes.