Gluing for Type Theory

Ambrus Kaposi
Eötvös Loránd University, Budapest, Hungary
akaposi@inf.elte.hu

Simon Huber
University of Gothenburg, Sweden
simonhu@chalmers.se

Christian Sattler
University of Gothenburg, Sweden
sattler@chalmers.se

Abstract

The relationship between categorical gluing and proofs using the logical relation technique is folklore. In this paper we work out this relationship for Martin-Löf type theory and show that parametricity and canonicity arise as special cases of gluing. The input of gluing is two models of type theory and a pseudomorphism between them and the output is a displayed model over the first model. A pseudomorphism preserves the categorical structure strictly, the empty context and context extension up to isomorphism, and there are no conditions on preservation of type formers. We look at three examples of pseudomorphisms: the identity on the syntax, the interpretation into the set model and the global section functor. Gluing along these result in syntactic parametricity, semantic parametricity and canonicity, respectively.

2012 ACM Subject Classification Theory of computation → Type theory

Keywords and phrases Martin-Löf type theory, logical relations, parametricity, canonicity, quotient inductive types

Digital Object Identifier 10.4230/LIPIcs.FSCD.2019.25

Funding *Ambrus Kaposi*: The author was supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications) and COST Action EUTypes CA15123. *Simon Huber*: I acknowledge the support of the Centre for Advanced Study (CAS) in Oslo, Norway, which funded and hosted the research project Homotopy Type Theory and Univalent Foundations during the academic year 2018/19.

Acknowledgements The authors thank Thorsten Altenkirch, Simon Boulier, Thierry Coquand, András Kovács and Nicolas Tabareau for discussions related to the topics of this paper.

1 Introduction

Categorical gluing [11, Section 4.10] is a method to form a new category from two categories and a functor between them. This goes back to the Artin gluing of Grothendieck toposes [5, Exposé IV, Section 9.5]. Given a functor F from category $\mathcal{S}$ to $\mathcal{M}$, an object in the glued category is a triple $\Gamma : |\mathcal{S}|$, $\Delta : |\mathcal{M}|$ and a morphism $\mathcal{M}(\Delta, F\,\Gamma)$. Models of logics and type theories can be given as categories with extra structure and gluing can be extended to these models. Gluing was used to prove properties of closed proofs in intuitionistic higher-order logic [21] and normalisation for simple type theory [14, 26] and System F [2]. In programming language semantics, similar results are proved more syntactically using the technique of logical relations [22, 24], see [15] for an introduction and [13, 1] for more recent proofs using this technique. It is folklore that logical relations correspond to gluing. Logical relations scale to real-world systems [27, 19] while gluing is a more abstract construction which can be applied to systems with well-understood categorical semantics.

4th International Conference on Formal Structures for Computation and Deduction (FSCD 2019).
Editor: Herman Geuvers; Article No. 25; pp. 25:1–25:19
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In this paper we develop the correspondence between proof-relevant logical predicates and gluing for Martin-Löf type theory. Logical relations were defined for type theory to prove free theorems in syntactic [7] and semantic (Reynolds-style) [6] ways. Proof-relevant logical predicates were employed to prove normalisation and canonicity for type theory [4, 10, 20]. We unify these approaches by defining gluing in an abstract way, for any pseudomorphism between two models of type theory. An important characteristic of our approach is using an algebraic syntax of type theory. By this we mean the well-typed syntax of type theory given as a quotient inductive-inductive type (QIIT, [18]). A model of this syntax is just an algebra of the QIIT which turns out to be the same as a category with families (CwF, [12]) with extra structure. A pseudomorphism of models is a map from sorts in one model to sorts in the other model which preserves the categorical structure strictly and the empty context and context extension up to isomorphism. We show that gluing can be performed along any pseudomoprhism and gluing preserves Π, Σ, Bool and an infinite hierarchy of universes.

Our motivational guideline for this paper is the following.

1. Gluing over identity is syntactic parametricity.
2. Gluing over the interpretation into the set model is semantic parametricity.
3. Gluing over the global section functor is canonicity.
4. Gluing over Yoneda is normalisation.
5. Gluing over Yoneda composed with the set interpretation is definability/completeness.

In this paper we only generalise steps 1–3. The Yoneda embedding (from the syntax to the presheaf model over a wide subcategory of contexts and substitutions) is also a pseudomorphism, so our paper applies to steps 4–5 as well. However, Yoneda has extra structure that we do not employ in this paper. This extra structure is needed to obtain full normalisation or completeness.

Structure of the paper

After summarizing related work and the metatheory, we define our object type theory in Section 2 and as an example we define its set model (Section 3). Then we define the notion of pseudomorphism (Section 4) and gluing for any pseudomorphism (Section 5). Afterwards, in Section 6 we define a non-trivial pseudomorphism: the global section functor which goes from the syntax to the set model and maps types to terms of the type in the empty context. We put together the pieces in Section 7 by obtaining parametricity and canonicity for our object theory using gluing. We conclude and summarize further work in Section 8.

Contribution

The contribution of this paper is showing that gluing can be defined for any pseudomorphism for Martin-Löf type theory. To our knowledge, this is the first general construction from which both parametricity and canonicity arise.

Related work

Sterling and Spitters [26] developed gluing for simple type theory and show how it relates to syntactic proofs of normalisation by logical relations and semantic proofs based on normalisation by evaluation. Altenkirch, Hofmann and Streicher developed gluing for System F and prove normalisation in their unpublished note [2]. Rabe and Sojakova [23] defined a syntactic framework for logical relations which applies to theories formulated in the Edinburgh

Logical Framework (LF). Shulman [25] developed gluing for type theory in the context of type-theoretic fibration categories and proves homotopy canonicity for a 1-truncated version of homotopy type theory. Compared to Shulman, we work with a notion of model closer to the syntax of type theory: categories with families. In previous work [4] we proved normalisation for type theory with Π, a base type and a base family. The logical predicate used in that proof is an instance of the abstract gluing technique presented in this paper. Coquand [10] proves canonicity and normalisation for a richer type theory with Bool and a hierarchy of universes. His canonicity proof is an unfolding of the canonicity proof given in this paper.

Metatheory and notation

Our metatheory is Martin-Löf's extensional type theory. We have a cumulative hierarchy of universes Set_0, Set_1, ... with Set_ω on top. Sometimes we omit the universe indices. Function space is denoted by $\to$ with constructor λ and application written as juxtaposition. We use implicit arguments extensively, e.g. we would write the type of function composition as $(B \to C) \to (A \to B) \to (A \to C)$ instead of $(A : \mathsf{Set}) \to (B : \mathsf{Set}) \to (C : \mathsf{Set}) \to (B \to C) \to (A \to B) \to (A \to C)$. When a metavariable is not quantified explicitly (such as A, B, C), we assume implicit quantification and implicit application as well. Sometimes we omit explicit arguments for readability, in this case we write underscore _ instead of the argument. Pairs are denoted by $\times$ with constructor $-,-$ and destructors $._1$ and $._2$. Both $\to$ and $\times$ come with η laws. The one-element type is denoted $\mathbb{1}$ with constructor $*$, the two-element type is denoted $\mathbb{2}$, its constructors being $*$ and $**$ and its eliminator case. Equality is denoted $=$ and we use equational reasoning to write equality proofs. We use equality reflection in the metatheory (not in our object theory described in Section 2). Following Hofmann [16], our arguments can be translated to an intensional type theory with function extensionality and uniqueness of identity proofs.

2 Type theory

By *type theory* we mean the (generalised) algebraic structure in Figure 1 with four sorts, 26 operators, and 34 equations. The four sorts are those of contexts, types, substitutions, and terms. Contexts and types are indexed by a universe level which is a metatheoretic natural number. Furthermore, types are indexed by contexts and terms by a context and a type in that context so that we can only mention well-typed terms. Substitutions are indexed by their domain and codomain, both contexts.

We explain the operators and laws for the substitution calculus (first column, operators id to $,\circ$) as follows: Con and Sub form a category (id to idr); there is a contravariant, functorial action of substitutions on types and terms ($-[-]$ to $[\circ]$), thus types (of fixed level) form a presheaf on the category and terms form a presheaf on its category of elements; there is an empty context $\cdot$ with a unique ($\cdot\eta$) empty substitution ϵ into it, thus $\cdot$ is the terminal object of the category; extended contexts can be formed using $- \triangleright -$ and there is a natural isomorphism between substitutions into $\Delta \triangleright A$ and a pair of a substitution σ into Δ and a term of type $A[\sigma]$ ($-,-$ to $,\circ$). The substitution calculus is the same as the structure of a category with families (CwF, [12]) with contexts and types indexed over natural numbers representing universe levels. In the CwF language, context extension is called comprehension. We denote n-fold iteration of the weakening substitution p by p^n (where $\mathsf{p}^0 = \mathsf{id}$), and we denote De Bruijn indices by natural numbers, i.e. $0 := \mathsf{q}$, $1 := \mathsf{q}[\mathsf{p}]$, ..., $n := \mathsf{q}[\mathsf{p}^n]$. We define

Con	$: \mathbb{N} \to \mathsf{Set}$
Ty	$: \mathbb{N} \to \mathsf{Con}\,i \to \mathsf{Set}$
Sub	$: \mathsf{Con}\,i \to \mathsf{Con}\,j \to \mathsf{Set}$
Tm	$: (\Gamma : \mathsf{Con}\,i) \to \mathsf{Ty}\,j\,\Gamma \to \mathsf{Set}$
id	$: \mathsf{Sub}\,\Gamma\,\Gamma$
$-\circ-$	$: \mathsf{Sub}\,\Theta\,\Delta \to \mathsf{Sub}\,\Gamma\,\Theta \to \mathsf{Sub}\,\Gamma\,\Delta$
ass	$: (\sigma \circ \delta) \circ \nu = \sigma \circ (\delta \circ \nu)$
idl	$: \mathsf{id} \circ \sigma = \sigma$
idr	$: \sigma \circ \mathsf{id} = \sigma$
$-[-]$	$: \mathsf{Ty}\,i\,\Delta \to \mathsf{Sub}\,\Gamma\,\Delta \to \mathsf{Ty}\,i\,\Gamma$
$-[-]$	$: \mathsf{Tm}\,\Delta\,A \to (\sigma : \mathsf{Sub}\,\Gamma\,\Delta) \to$ $\mathsf{Tm}\,\Gamma\,(A[\sigma])$
$[\mathsf{id}]$	$: A[\mathsf{id}] = A$
$[\circ]$	$: A[\sigma \circ \delta] = A[\sigma][\delta]$
$[\mathsf{id}]$	$: t[\mathsf{id}] = t$
$[\circ]$	$: t[\sigma \circ \delta] = t[\sigma][\delta]$
$\cdot$	$: \mathsf{Con}\,0$
ϵ	$: \mathsf{Sub}\,\Gamma\,\cdot$
$\cdot\eta$	$: (\sigma : \mathsf{Sub}\,\Gamma\,\cdot) = \epsilon$
$-\triangleright-$	$: (\Gamma : \mathsf{Con}\,i) \to \mathsf{Ty}\,j\,\Gamma \to \mathsf{Con}\,(i \sqcup j)$
$-,-$	$: (\sigma : \mathsf{Sub}\,\Gamma\,\Delta) \to \mathsf{Tm}\,\Gamma\,(A[\sigma]) \to$ $\mathsf{Sub}\,\Gamma\,(\Delta \triangleright A)$
p	$: \mathsf{Sub}\,(\Gamma \triangleright A)\,\Gamma$
q	$: \mathsf{Tm}\,(\Gamma \triangleright A)\,(A[\mathsf{p}])$
$\triangleright\beta_1$	$: \mathsf{p} \circ (\sigma, t) = \sigma$
$\triangleright\beta_2$	$: \mathsf{q}[\sigma, t] = t$
$\triangleright\eta$	$: (\mathsf{p}, \mathsf{q}) = \mathsf{id}$
$,\circ$	$: (\sigma, t) \circ \nu = (\sigma \circ \nu, t[\nu])$
Π	$: (A : \mathsf{Ty}\,i\,\Gamma) \to \mathsf{Ty}\,j\,(\Gamma \triangleright A) \to$ $\mathsf{Ty}\,(i \sqcup j)\,\Gamma$
lam	$: \mathsf{Tm}\,(\Gamma \triangleright A)\,B \to \mathsf{Tm}\,\Gamma\,(\Pi\,A\,B)$
app	$: \mathsf{Tm}\,\Gamma\,(\Pi\,A\,B) \to \mathsf{Tm}\,(\Gamma \triangleright A)\,B$
$\Pi\beta$	$: \mathsf{app}\,(\mathsf{lam}\,t) = t$
$\Pi\eta$	$: \mathsf{lam}\,(\mathsf{app}\,t) = t$
$\Pi[]$	$: (\Pi\,A\,B)[\sigma] = \Pi\,(A[\sigma])\,(B[\sigma^{\uparrow}])$
lam[]	$: (\mathsf{lam}\,t)[\sigma] = \mathsf{lam}\,(t[\sigma^{\uparrow}])$

Σ	$: (A : \mathsf{Ty}\,i\,\Gamma) \to \mathsf{Ty}\,j\,(\Gamma \triangleright A) \to$ $\mathsf{Ty}\,(i \sqcup j)\,\Gamma$
$-,-$	$: (u : \mathsf{Tm}\,\Gamma\,A) \to \mathsf{Tm}\,\Gamma\,(B[\mathsf{id}, u]) \to$ $\mathsf{Tm}\,\Gamma\,(\Sigma\,A\,B)$
projl	$: \mathsf{Tm}\,\Gamma\,(\Sigma\,A\,B) \to \mathsf{Tm}\,\Gamma\,A$
projr	$: (t : \mathsf{Tm}\,\Gamma\,(\Sigma\,A\,B)) \to$ $\mathsf{Tm}\,\Gamma\,(B[\mathsf{id}, \mathsf{projl}\,t])$
$\Sigma\beta_1$	$: \mathsf{projl}\,(u, v) = u$
$\Sigma\beta_2$	$: \mathsf{projr}\,(u, v) = v$
$\Sigma\eta$	$: (\mathsf{projl}\,t, \mathsf{projr}\,t) = t$
$\Sigma[]$	$: (\Sigma\,A\,B)[\sigma] = \Sigma\,(A[\sigma])\,(B[\sigma^{\uparrow}])$
$,[]$	$: (u, v)[\sigma] = (u[\sigma], v[\sigma])$
$\top$	$: \mathsf{Ty}\,0\,\Gamma$
tt	$: \mathsf{Tm}\,\Gamma\,\top$
$\top\eta$	$: (t : \mathsf{Tm}\,\Gamma\,\top) = \mathsf{tt}$
$\top[]$	$: \top[\sigma] = \top$
tt[]	$: \mathsf{tt}[\sigma] = \mathsf{tt}$
U	$: (i : \mathbb{N}) \to \mathsf{Ty}\,(i+1)\,\Gamma$
El	$: \mathsf{Tm}\,\Gamma\,(\mathsf{U}\,i) \to \mathsf{Ty}\,i\,\Gamma$
c	$: \mathsf{Ty}\,i\,\Gamma \to \mathsf{Tm}\,\Gamma\,(\mathsf{U}\,i)$
$\mathsf{U}\beta$	$: \mathsf{El}\,(\mathsf{c}\,A) = A$
$\mathsf{U}\eta$	$: \mathsf{c}\,(\mathsf{El}\,a) = a$
U[]	$: (\mathsf{U}\,i)[\sigma] = (\mathsf{U}\,i)$
El[]	$: (\mathsf{El}\,a)[\sigma] = \mathsf{El}\,(a[\sigma])$
Bool	$: \mathsf{Ty}\,0\,\Gamma$
true	$: \mathsf{Tm}\,\Gamma\,\mathsf{Bool}$
false	$: \mathsf{Tm}\,\Gamma\,\mathsf{Bool}$
if	$: (C : \mathsf{Ty}\,i\,(\Gamma \triangleright \mathsf{Bool})) \to$ $\mathsf{Tm}\,\Gamma\,(C[\mathsf{id}, \mathsf{true}]) \to$ $\mathsf{Tm}\,\Gamma\,(C[\mathsf{id}, \mathsf{false}]) \to$ $(t : \mathsf{Tm}\,\Gamma\,\mathsf{Bool}) \to \mathsf{Tm}\,\Gamma\,(C[\mathsf{id}, t])$
$\mathsf{Bool}\beta_1$	$: \mathsf{if}\,C\,u\,v\,\mathsf{true} = u$
$\mathsf{Bool}\beta_2$	$: \mathsf{if}\,C\,u\,v\,\mathsf{false} = v$
Bool[]	$: \mathsf{Bool}[\sigma] = \mathsf{Bool}$
true[]	$: \mathsf{true}[\sigma] = \mathsf{true}$
false[]	$: \mathsf{false}[\sigma] = \mathsf{false}$
if[]	$: (\mathsf{if}\,C\,u\,v\,t)[\sigma] =$ $\mathsf{if}\,(C[\sigma^{\uparrow}])\,(u[\sigma])\,(v[\sigma])\,(t[\sigma])$

Figure 1 Type theory as a generalised algebraic structure. $\sigma^{\uparrow}$ abbreviates $(\sigma \circ \mathsf{p}, \mathsf{q})$.

lifting of a substitution $\sigma : \mathsf{Sub}\,\Gamma\,\Delta$ by $\sigma^{\uparrow} : \mathsf{Sub}\,(\Gamma \triangleright A[\sigma])\,(\Delta \triangleright A) := (\sigma \circ \mathsf{p}, \mathsf{q})$. We observe that it has the property ${}^{\uparrow}[] : (\sigma^{\uparrow})[\delta, t] = (\sigma \circ \delta, t)$.

Π-types are characterized by a natural isomorphism between $\mathsf{Tm}\,\Gamma\,(\Pi\,A\,B)$ and $\mathsf{Tm}\,(\Gamma \triangleright A)\,B$ (lam and app). We define the usual application as $t\,\$\,u := (\mathsf{app}\,t)[\mathsf{id}, u]$. $A \Rightarrow B$ abbreviates $\Pi\,A\,(B[\mathsf{p}])$. Σ-types are given by the constructor $-,-$ and projections projl and projr and they support an η-law. There is a unit type $\top$ with one constructor tt and an η-law and there is a hierarchy of Tarski-universes, given by natural isomorphisms between $\mathsf{Ty}\,i\,\Gamma$ and $\mathsf{Tm}\,\Gamma\,(\mathsf{U}\,i)$ for every i.[1] As terms of Π-, Σ- and U-types are characterized by natural isomorphisms, we stated the substitution law for only one of the two directions, the other can be derived. We illustrate how to do this for app and state the other laws.

$$
\begin{aligned}
&\mathsf{app}[] &&: (\mathsf{app}\,t)[\sigma^{\uparrow}] \overset{\Pi\beta}{=} \mathsf{app}\,(\mathsf{lam}\,((\mathsf{app}\,t)[\sigma^{\uparrow}])) \overset{\mathsf{lam}[]}{=} \mathsf{app}\,((\mathsf{lam}\,(\mathsf{app}\,t))[\sigma]) \overset{\Pi\eta}{=} \mathsf{app}\,(t[\sigma]) \\
&\$[] &&: (t\,\$\,u)[\sigma] = t[\sigma]\,\$\,u[\sigma] \\
&\mathsf{projl}[] &&: (\mathsf{projl}\,t)[\sigma] = \mathsf{projl}\,(t[\sigma]) \\
&\mathsf{projr}[] &&: (\mathsf{projr}\,t)[\sigma] = \mathsf{projr}\,(t[\sigma]) \\
&\mathsf{c}[] &&: (c\,A)[\sigma] = c\,(A[\sigma])
\end{aligned}
$$

Finally, we have booleans with a dependent eliminator if into any universe. Sometimes for readability we omit the first argument (C) of if and write _ instead.

Note that the well-typedness of some of the equations depends on previous equations. For example, the left-hand side of $\triangleright\beta_2$ has type $\mathsf{Tm}\,\Gamma\,(A[\mathsf{p}][\sigma, t])$, while the right-hand side has $\mathsf{Tm}\,\Gamma\,(A[\sigma])$, and these types are equal by $[\circ]$ and $\triangleright\beta_1$. In an intensional metatheory, we would need to transport the left-hand side over these equations. We use an extensional metatheory, so we do not write such dependencies.

As an example we write the polymorphic identity function as $\mathsf{lam}\,(\mathsf{lam}\,\mathsf{q})$. Note that lam and q have several implicit arguments that we did not write down. However, when we write a term, these implicit arguments should be clear from the context. In this example, saying that it has type $\mathsf{Tm}\,\cdot\,(\Pi\,(\mathsf{U}\,0)\,(\mathsf{El}\,\mathsf{q} \Rightarrow \mathsf{El}\,\mathsf{q}))$ fixes all its implicit arguments. We don't have raw terms with a type assignment or type inference system, we only work with fully annotated well-typed terms where lots of information is implicit (as usual in mathematics). This is sometimes called intrinsic or well-typed syntax, see [3] for an introduction.

We call algebras of the algebraic structure presented in Figure 1 *models* of type theory. When referring to different models, we put the model as a subscript, i.e. $\mathsf{Con}_{\mathcal{M}}$ refers to contexts in model $\mathcal{M}$, $\mathsf{id}_{\mathcal{M}} : \mathsf{Sub}_{\mathcal{M}}\,\Gamma_{\mathcal{M}}\,\Gamma_{\mathcal{M}}$ refers to the identity substitution in this model. For metavariables, we usually use the same subscript as for the two occurrences of $\Gamma_{\mathcal{M}}$ in the type of $\mathsf{id}_{\mathcal{M}}$.

We introduce some basic notions for working with models. These notions are obtained mechanically by viewing Figure 1 as a scheme for a quotient inductive-inductive type (QIIT, [18]).

[1] We learned this representation of universes from Thierry Coquand. Note that Russell universes could be represented by replacing El and c by the sort equation $\mathsf{Ty}\,i\,\Gamma = \mathsf{Tm}\,\Gamma\,(\mathsf{U}\,i)$ and the equation $A[\sigma]_{\mathsf{Ty}} = A[\sigma]_{\mathsf{Tm}}$ relating type and term substitutions. The latter equation is well-typed because of the former.

- A (strict) *morphism* H between models $\mathcal{M}$ and $\mathcal{N}$ consists of four functions between the sorts which preserve all the 26 operators (up to equality). We use subscripts to mark which component we mean, e.g. some of the components are the following.

$$\begin{aligned}
&H_{\mathsf{Con}} &&: \mathsf{Con}_{\mathcal{M}}\, i \to \mathsf{Con}_{\mathcal{N}}\, i\\
&H_{\mathsf{Ty}} &&: \mathsf{Ty}_{\mathcal{M}}\, j\, \Gamma \to \mathsf{Ty}_{\mathcal{N}}\, j\, (H\, \Gamma)\\
&H_{\mathsf{Sub}} &&: \mathsf{Sub}_{\mathcal{M}}\, \Gamma\, \Delta \to \mathsf{Sub}_{\mathcal{N}}\, (H\, \Gamma)\, (H\, \Delta)\\
&H_{\mathsf{Tm}} &&: \mathsf{Tm}_{\mathcal{M}}\, \Gamma\, A \to \mathsf{Tm}_{\mathcal{N}}\, (H\, \Gamma)\, (H\, A)\\
&H_{[]} &&: H_{\mathsf{Ty}}\, (A[\sigma]_{\mathcal{M}}) = (H_{\mathsf{Ty}}\, A)[H_{\mathsf{Sub}}\, \sigma]_{\mathcal{N}}\\
&H_{\triangleright} &&: H_{\mathsf{Con}}\, (\Gamma \triangleright_{\mathcal{M}} A) = H_{\mathsf{Con}}\, \Gamma \triangleright_{\mathcal{N}} H_{\mathsf{Ty}}\, A\\
&H_{\Pi} &&: H_{\mathsf{Ty}}\, (\Pi_{\mathcal{M}}\, A\, B) = \Pi_{\mathcal{N}}\, (H_{\mathsf{Ty}}\, A)\, (H_{\mathsf{Ty}}\, B)\\
&H_{\mathsf{lam}} &&: H_{\mathsf{Tm}}\, (\mathsf{lam}_{\mathcal{M}}\, t) = \mathsf{lam}_{\mathcal{N}}\, (H_{\mathsf{Tm}}\, t)\\
&H_{\mathsf{app}} &&: H_{\mathsf{Tm}}\, (\mathsf{app}_{\mathcal{M}}\, t) = \mathsf{app}_{\mathcal{N}}\, (H_{\mathsf{Tm}}\, t)
\end{aligned}$$

Sometimes we omit subscripts for readability, e.g. above we wrote $H\,\Gamma$ instead of $H_{\mathsf{Con}}\,\Gamma$ and we also did not decorate metavariables with subscripts, all Γ above live in $\mathsf{Con}_{\mathcal{M}}$, all σ in $\mathsf{Sub}_{\mathcal{M}}$ etc. We will follow this convention later.

- A *displayed model* $\mathcal{Q}$ over a model $\mathcal{M}$ encodes a model with a strict morphism to $\mathcal{M}$. It is given by four families, 26 operations, and 34 equalities, all of which are over those of $\mathcal{M}$, e.g.

$$\begin{aligned}
&\mathsf{Con}_{\mathcal{Q}} &&: (i : \mathbb{N}) \to \mathsf{Con}_{\mathcal{M}}\, i \to \mathsf{Set}\\
&\mathsf{Ty}_{\mathcal{Q}} &&: (j : \mathbb{N}) \to \mathsf{Con}_{\mathcal{Q}}\, i\, \Gamma \to \mathsf{Ty}_{\mathcal{M}}\, j\, \Gamma \to \mathsf{Set}\\
&\mathsf{Sub}_{\mathcal{Q}} &&: \mathsf{Con}_{\mathcal{Q}}\, i\, \Gamma \to \mathsf{Con}_{\mathcal{Q}}\, j\, \Delta \to \mathsf{Sub}_{\mathcal{M}}\, \Gamma\, \Delta \to \mathsf{Set}\\
&\mathsf{Tm}_{\mathcal{Q}} &&: (\Gamma_{\mathcal{Q}} : \mathsf{Con}_{\mathcal{Q}}\, i\, \Gamma) \to \mathsf{Ty}_{\mathcal{Q}}\, j\, \Gamma_{\mathcal{Q}}\, A \to \mathsf{Tm}_{\mathcal{M}}\, \Gamma\, A \to \mathsf{Set}\\
&-[-]_{\mathcal{Q}} &&: \mathsf{Ty}_{\mathcal{Q}}\, j\, \Delta_{\mathcal{Q}}\, A \to \mathsf{Sub}_{\mathcal{Q}}\, \Gamma_{\mathcal{Q}}\, \Delta_{\mathcal{Q}}\, \sigma \to \mathsf{Ty}_{\mathcal{Q}}\, j\, \Gamma_{\mathcal{Q}}\, (A[\sigma]_{\mathcal{M}})\\
&- \triangleright_{\mathcal{Q}} - &&: (\Gamma_{\mathcal{Q}} : \mathsf{Con}_{\mathcal{Q}}\, i\, \Gamma) \to \mathsf{Ty}_{\mathcal{Q}}\, j\, \Gamma_{\mathcal{Q}}\, A \to \mathsf{Con}_{\mathcal{Q}}\, (i \sqcup j)\, (\Gamma \triangleright_{\mathcal{M}} A)\\
&\Pi_{\mathcal{Q}} &&: (A_{\mathcal{Q}} : \mathsf{Ty}_{\mathcal{Q}}\, i\, \Gamma_{\mathcal{Q}}\, A) \to \mathsf{Ty}_{\mathcal{Q}}\, j\, (\Gamma_{\mathcal{Q}} \triangleright_{\mathcal{Q}} A_{\mathcal{Q}})\, B \to \mathsf{Ty}_{\mathcal{Q}}\, (i \sqcup j)\, \Gamma_{\mathcal{Q}}\, (\Pi_{\mathcal{M}}\, A\, B)\\
&\mathsf{lam}_{\mathcal{Q}} &&: \mathsf{Tm}_{\mathcal{Q}}\, (\Gamma_{\mathcal{Q}} \triangleright_{\mathcal{Q}} A_{\mathcal{Q}})\, B_{\mathcal{Q}}\, t \to \mathsf{Tm}_{\mathcal{Q}}\, \Gamma_{\mathcal{Q}}\, (\Pi_{\mathcal{Q}}\, A_{\mathcal{Q}}\, B_{\mathcal{Q}})\, (\mathsf{lam}_{\mathcal{M}}\, t)\\
&\mathsf{app}_{\mathcal{Q}} &&: \mathsf{Tm}_{\mathcal{Q}}\, \Gamma_{\mathcal{Q}}\, (\Pi_{\mathcal{Q}}\, A_{\mathcal{Q}}\, B_{\mathcal{Q}})\, t \to \mathsf{Tm}_{\mathcal{Q}}\, (\Gamma_{\mathcal{Q}} \triangleright_{\mathcal{Q}} A_{\mathcal{Q}})\, B_{\mathcal{Q}}\, (\mathsf{app}_{\mathcal{M}}\, t)\\
&\Pi\beta_{\mathcal{Q}} &&: \mathsf{app}_{\mathcal{Q}}\, (\mathsf{lam}_{\mathcal{Q}}\, t_{\mathcal{Q}}) = t_{\mathcal{Q}}
\end{aligned}$$

- A *section* I of a displayed model $\mathcal{Q}$ over $\mathcal{M}$ is like a dependent morphism, encoding a section to the strict morphism to $\mathcal{M}$ encoded by $\mathcal{Q}$. It contains, among others, the following components.

$$\begin{aligned}
&I_{\mathsf{Con}} &&: (\Gamma : \mathsf{Con}_{\mathcal{M}}\, i) \to \mathsf{Con}_{\mathcal{Q}}\, i\, \Gamma\\
&I_{\mathsf{Ty}} &&: (A : \mathsf{Ty}_{\mathcal{M}}\, j\, \Gamma) \to \mathsf{Ty}_{\mathcal{Q}}\, j\, (I\, \Gamma)\, A\\
&I_{\mathsf{Sub}} &&: (\sigma : \mathsf{Sub}_{\mathcal{M}}\, \Gamma\, \Delta) \to \mathsf{Sub}_{\mathcal{Q}}\, (I\, \Gamma)\, (I\, \Delta)\, \sigma\\
&I_{A[\sigma]} &&: I\, (A[\sigma]_{\mathcal{M}}) = (I\, A)[I\, \sigma]_{\mathcal{Q}}\\
&I_{\triangleright} &&: I\, (\Gamma \triangleright_{\mathcal{M}} A) = I\, \Gamma \triangleright_{\mathcal{Q}} I\, A\\
&I_{\Pi} &&: I\, (\Pi_{\mathcal{M}}\, A\, B) = \Pi_{\mathcal{Q}}\, (I\, A)\, (I\, B)\\
&I_{\mathsf{lam}} &&: I\, (\mathsf{lam}_{\mathcal{M}}\, t) = \mathsf{lam}_{\mathcal{Q}}\, (I\, t)\\
&I_{\mathsf{app}} &&: I\, (\mathsf{app}_{\mathcal{M}}\, t) = \mathsf{app}_{\mathcal{Q}}\, (I\, t)
\end{aligned}$$

We assume the existence of the quotient inductive-inductive type (QIIT, [18]) specified by Figure 1. We thus have an initial model S, called the *syntax*. For every model $\mathcal{M}$, the *recursor* $\mathsf{rec}^{\mathcal{M}}$ is the unique morphism from S to $\mathcal{M}$. For every displayed model $\mathcal{Q}$ over S, the *eliminator* $\mathsf{elim}^{\mathcal{Q}}$ is the unique section of $\mathcal{Q}$.

2.1 The identity type

In our construction of gluing we will assume that the target model has identity types. Identity types extend type theory as given in Figure 1 with the following operators and equations.

$$\begin{aligned}
&\mathsf{Id} &&: (A : \mathsf{Ty}\,i\,\Gamma) \to \mathsf{Tm}\,\Gamma\,A \to \mathsf{Tm}\,\Gamma\,A \to \mathsf{Ty}\,i\,\Gamma \\
&\mathsf{refl} &&: (u : \mathsf{Tm}\,\Gamma\,A) \to \mathsf{Tm}\,\Gamma\,(\mathsf{Id}\,A\,u\,u) \\
&\mathsf{J} &&: \big(C : \mathsf{Ty}\,i\,(\Gamma \triangleright A \triangleright \mathsf{Id}\,(A[\mathsf{p}])\,(u[\mathsf{p}])\,0)\big) \to \mathsf{Tm}\,\Gamma\,(C[\mathsf{id}, u, \mathsf{refl}\,u]) \to \\
& &&\quad (e : \mathsf{Tm}\,\Gamma\,(\mathsf{Id}\,A\,u\,v)) \to \mathsf{Tm}\,\Gamma\,(C[\mathsf{id}, v, e[\mathsf{p}]]) \\
&\mathsf{Id}\beta &&: \mathsf{J}\,C\,w\,(\mathsf{refl}\,u) = w \\
&\mathsf{Id}[] &&: (\mathsf{Id}\,A\,u\,v)[\sigma] = \mathsf{Id}\,(A[\sigma])\,(u[\sigma])\,(v[\sigma]) \\
&\mathsf{refl}[] &&: (\mathsf{refl}\,u)[\sigma] = \mathsf{refl}\,(u[\sigma]) \\
&\mathsf{J}[] &&: (\mathsf{J}\,C\,w\,e)[\sigma] = \mathsf{J}\,(C[\sigma^{\uparrow\uparrow}])\,(w[\sigma])\,(e[\sigma])
\end{aligned}$$

$\mathsf{Id}\,A\,u\,v$ expresses that u is equal to v, there is one constructor refl expressing reflexivity and there is the eliminator J which says that given a family over identities and a witness of that family for refl we get that there is an element of that family for every identity proof.

3 The Set model

As an example of a simple model, we define the set model (standard model, metacircular model). In this model, contexts are sets, types are families over their contexts, substitutions are functions, and terms are dependent functions. Context extension is metatheoretic Σ, otherwise everything is modelled by its metatheoretic counterparts, e.g. Π-types are dependent functions, lam is λ, app is metatheoretic application. We list a few components for illustration.

$$\begin{aligned}
&\mathsf{Con}\,i &&:= \mathsf{Set}_i \\
&\mathsf{Ty}\,j\,\Gamma &&:= \Gamma \to \mathsf{Set}_j \\
&\mathsf{Sub}\,\Gamma\,\Delta &&:= \Gamma \to \Delta \\
&\mathsf{Tm}\,\Gamma\,A &&:= (\gamma : \Gamma) \to A\,\gamma \\
&A[\sigma] &&:= \lambda\gamma.A\,(\sigma\,\gamma) \\
&\cdot &&:= \mathbb{1} \\
&\epsilon &&:= \lambda\_.* \\
&\Gamma \triangleright A &&:= (\gamma : \Gamma) \times A\,\gamma \\
&(\sigma, t) &&:= (\sigma, t) \\
&\mathsf{p} &&:= \mathsf{projl} \\
&\mathsf{q} &&:= \mathsf{projr} \\
&\Pi\,A\,B &&:= \lambda\gamma.(\alpha : A\,\gamma) \to B\,(\gamma, \alpha)
\end{aligned}$$

$$\begin{aligned}
&\mathsf{lam}\,t && := \lambda\gamma.\lambda\alpha.t\,(\gamma,\alpha)\\
&\mathsf{app}\,t && := \lambda\gamma.t\,\gamma_{.1}\,\gamma_{.2}\\
&\Pi\beta && : \mathsf{app}\,(\mathsf{lam}\,t) = \lambda\gamma'.(\lambda\gamma.\lambda\alpha.t\,(\gamma,\alpha))\,\gamma'_{.1}\,\gamma'_{.2} \overset{\rightarrow\beta}{=} \lambda\gamma'.t\,(\gamma'_{.1},\gamma'_{.2}) \overset{\times\eta}{=} \lambda\gamma'.t\,\gamma' \overset{\rightarrow\eta}{=} t\\
&\mathsf{U}\,i && := \lambda\_.\mathsf{Set}_i\\
&\mathsf{El}\,a && := a\\
&\mathsf{c}\,a && := a\\
&\mathsf{Bool} && := \mathbb{2}\\
&\mathsf{true} && := *\\
&\mathsf{false} && := **\\
&\mathsf{if}\,C\,t\,u\,v && := \mathsf{case}\,t\,u\,v\\
&\mathsf{Id}\,A\,u\,v && := (u = v)
\end{aligned}$$

The β-law for Π uses the metatheoretic β- and η-laws for the functions and η for pairs.

Using the recursor we can define an interpreter for our syntax which maps syntactic terms to metatheoretic objects.

$$\begin{aligned}
&[\![-]\!] : \mathsf{Con}_{\mathsf{S}}\,i && \to \mathsf{Set}_i && := \mathsf{rec}^{\mathsf{Set}}_{\mathsf{Con}}\\
&[\![-]\!] : \mathsf{Ty}_{\mathsf{S}}\,j\,\Gamma && \to [\![\Gamma]\!] \to \mathsf{Set}_j && := \mathsf{rec}^{\mathsf{Set}}_{\mathsf{Ty}}\\
&[\![-]\!] : \mathsf{Sub}_{\mathsf{S}}\,\Gamma\,\Delta && \to [\![\Gamma]\!] \to [\![\Delta]\!] && := \mathsf{rec}^{\mathsf{Set}}_{\mathsf{Sub}}\\
&[\![-]\!] : \mathsf{Tm}_{\mathsf{S}}\,\Gamma\,A && \to (\gamma : [\![\Gamma]\!]) \to [\![A]\!]\,\gamma && := \mathsf{rec}^{\mathsf{Set}}_{\mathsf{Tm}}
\end{aligned}$$

For example, the interpretation of the polymorphic identity function is

$$[\![\mathsf{lam}\,(\mathsf{lam}\,\mathsf{q})]\!] : (\gamma : \mathbb{1}) \to (A : \mathsf{Set}_0) \to A \to A = \lambda\gamma.\lambda A.\lambda a.a.$$

4 Pseudomorphism

In this section we define morphisms of models of type theory which are strict on the category structure and weak on $\cdot$ and $-\triangleright-$. We call such a morphism a *pseudomorphism*, its components are listed in Figure 2. In terms of using comprehension categories [17] to describe models of type theory, this corresponds to the notion of morphism that relates the Grothendieck fibrations strictly, but the comprehension maps only up to natural isomorphism.

Just as a strict morphism (described in Section 2), a pseudomorphism F maps contexts in $\mathcal{S}$ to contexts in $\mathcal{M}$, types in $\mathcal{S}$ to types in $\mathcal{M}$, etc. Identity, composition and action on substitution are preserved strictly (F_{id}, $F_{\circ}$, $F_{[]}$ and $F_{[]}$). The empty context and context extension are preserved up to definitional isomorphism. Definitional isomorphism between two contexts $\Gamma : \mathsf{Con}\,i$, $\Delta : \mathsf{Con}\,j$ is defined as follows.

$$(f : \Gamma \cong \Delta) := (f_{.1} : \mathsf{Sub}\,\Gamma\,\Delta) \times (f_{.2} : \mathsf{Sub}\,\Delta\,\Gamma) \times (f_{.12} : f_{.1} \circ f_{.2} = \mathsf{id}) \times (f_{.21} : f_{.2} \circ f_{.1} = \mathsf{id})$$

$F_{\triangleright.1\circ}$ denotes a naturality condition that $F_{\triangleright}$ has to satisfy. The empty substitution ϵ and the comprehension operators $(-,-)$, p, q are preserved strictly, but this is up to the weakness of $\cdot$ and $\triangleright$.

Categorically, a pseudomorphism is a functor on categories of contexts with natural transformations on types and terms with the following properties: it preserves terminal objects; given $A : \mathsf{Ty}_{\mathcal{S}}\,j\,\Delta$, if a pair of $\sigma : \mathsf{Sub}_{\mathcal{S}}\,\Gamma\,\Delta$ and $t : \mathsf{Tm}_{\mathcal{S}}\,\Gamma\,(A[\sigma]_{\mathcal{S}})$ has the universal property of the context extension of Δ with A in $\mathcal{S}$, then the pair of $F\,\sigma$ and $F\,t$ has the universal property of the context extension of $F\,\Delta$ with $F\,A$ in $\mathcal{M}$.

$$
\begin{aligned}
&F_{\mathsf{Con}} &&: \mathsf{Con}_{\mathcal{S}}\, i \to \mathsf{Con}\, i\\
&F_{\mathsf{Ty}} &&: \mathsf{Ty}_{\mathcal{S}}\, j\, \Gamma \to \mathsf{Ty}\, j\, (F\,\Gamma)\\
&F_{\mathsf{Sub}} &&: \mathsf{Sub}_{\mathcal{S}}\, \Gamma\, \Delta \to \mathsf{Sub}\, (F\,\Gamma)\, (F\,\Delta)\\
&F_{\mathsf{Tm}} &&: \mathsf{Tm}_{\mathcal{S}}\, \Gamma\, A \to \mathsf{Tm}\, (F\,\Gamma)\, (F\,A)\\
&F_{\mathsf{id}} &&: F\, \mathsf{id}_{\mathcal{S}} = \mathsf{id}\\
&F_{\circ} &&: F\, (\sigma \circ_{\mathcal{S}} \delta) = F\,\sigma \circ F\,\delta\\
&F_{[]} &&: F\, (A[\sigma]_{\mathcal{S}}) = (F\,A)[F\,\sigma]\\
&F_{[]} &&: F\, (t[\sigma]_{\mathcal{S}}) = (F\,t)[F\,\sigma]\\
&F_{\cdot} &&: F\, \cdot_{\mathcal{S}} \cong \cdot\\
&F_{\epsilon} &&: F\, \epsilon_{\mathcal{S}} = F_{\cdot.2} \circ \epsilon\\
&F_{\triangleright} &&: F\, (\Gamma \triangleright_{\mathcal{S}} A) \cong F\,\Gamma \triangleright F\,A\\
&F_{\triangleright.1\circ} &&: F_{\triangleright.1} \circ F\, (\sigma^{\uparrow_{\mathcal{S}}}) = (F\,\sigma)^{\uparrow} \circ F_{\triangleright.1}\\
&F_{,} &&: F\, (\sigma ,_{\mathcal{S}} t) = F_{\triangleright.2} \circ (F\,\sigma, F\,t)\\
&F_{\mathsf{p}} &&: F\, \mathsf{p}_{\mathcal{S}} = \mathsf{p} \circ F_{\triangleright.1}\\
&F_{\mathsf{q}} &&: F\, \mathsf{q}_{\mathcal{S}} = \mathsf{q}[F_{\triangleright.1}]
\end{aligned}
$$

Figure 2 The components of a pseudomorphism F from $\mathcal{S}$ to $\mathcal{M}$. For readability, we omit the subscripts $_{\mathcal{S}}$ from the metavariable names and all the $_{\mathcal{M}}$ subscripts. That is, when we write Con or id we mean $\mathsf{Con}_{\mathcal{M}}$ and $\mathsf{id}_{\mathcal{M}}$. We overload the different parts of F, i.e. write F for F_{Con}, F_{Ty}, F_{Sub} and F_{Tm}. $\cong$ denotes definitional isomorphism, see in the text.

We derive the following naturality condition.

$$
\begin{aligned}
F_{\triangleright.2\circ} : F_{\triangleright.2} \circ_{\mathcal{M}} (F\,\sigma)^{\uparrow_{\mathcal{M}}} &\overset{F_{\triangleright.12}}{=} F_{\triangleright.2} \circ (F\,\sigma)^{\uparrow_{\mathcal{M}}} \circ F_{\triangleright.1} \circ F_{\triangleright.2} \overset{F_{\triangleright.1\circ}}{=}\\
&F_{\triangleright.2} \circ F_{\triangleright.1} \circ F\, (\sigma^{\uparrow_{\mathcal{S}}}) \circ F_{\triangleright.2} \overset{F_{\triangleright.21}}{=} F\, (\sigma^{\uparrow_{\mathcal{S}}}) \circ_{\mathcal{M}} F_{\triangleright.2}
\end{aligned}
$$

From this it follows that $F\, (\sigma^{\uparrow_{\mathcal{S}}}) = F_{\triangleright.2} \circ_{\mathcal{M}} (F\,\sigma)^{\uparrow_{\mathcal{M}}} \circ_{\mathcal{M}} F_{\triangleright.1}$.

Note that every strict morphism F is automatically pseudo, with $F_{\cdot.1} = F_{\cdot.2} = \mathsf{id}_{\mathcal{M}}$ and $F_{\triangleright.1} = F_{\triangleright.2} = \mathsf{id}_{\mathcal{M}}$.

A pseudomorphism automatically preserves type formers: Σ and $\top$ are preserved weakly, Π and U are preserved in a lax way and Bool in an oplax way. For example, we can define a map from $F\,(\Pi\,A\,B)$ to $\Pi\,(F\,A)\,(F\,B[F_{\triangleright.2}])$ as follows. We start using the eliminator of Π in $\mathcal{S}$ on a variable:

$$\mathsf{app}\,\mathsf{q} : \mathsf{Tm}_{\mathcal{S}}\, (\Gamma \triangleright \Pi\,A\,B \triangleright A[\mathsf{p}])\, (B[\mathsf{p}^{\uparrow}]).$$

Then we apply the pseudomorphism F and get a map in $\mathcal{M}$:

$$F\,(\mathsf{app}\,\mathsf{q}) : \mathsf{Tm}_{\mathcal{M}}\, \big(F\,(\Gamma \triangleright \Pi\,A\,B \triangleright A[\mathsf{p}])\big)\, \big(F\,(B[\mathsf{p}^{\uparrow}])\big).$$

Applying the substitution $F_{\triangleright.2} \circ (F_{\triangleright.2}{}^{\uparrow})$ and using the properties of F we obtain

$$F\,(\mathsf{app}\,\mathsf{q})[F_{\triangleright.2} \circ (F_{\triangleright.2}{}^{\uparrow})] : \mathsf{Tm}_{\mathcal{M}}\, \big(F\,\Gamma \triangleright F\,(\Pi\,A\,B) \triangleright F\,A[\mathsf{p}]\big)\, \big(F\,B[F_{\triangleright.2} \circ (\mathsf{p}^{\uparrow})]\big),$$

and finally we use the constructor of Π in $\mathcal{M}$:

$$\mathsf{lam}\, \big(F\,(\mathsf{app}\,\mathsf{q})[F_{\triangleright.2} \circ (F_{\triangleright.2}{}^{\uparrow})]\big) : \mathsf{Tm}_{\mathcal{M}}\, \big(F\,\Gamma \triangleright F\,(\Pi\,A\,B)\big)\, \big(\Pi\,(F\,A)\,(F\,B[F_{\triangleright.2}])\big).$$

We define a similar map for U, a map from Bool to $F\,\mathsf{Bool}$ and maps in both directions for Σ and $\top$. To express the latter we introduce the following abbreviation for a definitional isomorphism between types $A, B : \mathsf{Ty}\,\Gamma$.

$$\begin{aligned}(g : A \cong_\Gamma B) := {} & (g_{.1} : \mathsf{Tm}\,(\Gamma \triangleright A)\,(B[\mathsf{p}])) \times (g_{.2} : \mathsf{Tm}\,(\Gamma \triangleright B)\,(A[\mathsf{p}])) \times \\ & (g_{.12} : g_{.1}[\mathsf{p}, g_{.2}] = \mathsf{q}) \times (g_{.21} : g_{.2}[\mathsf{p}, g_{.1}] = \mathsf{q})\end{aligned}$$

For a pseudomorphism F the following comparison maps can be given.

$$\begin{aligned}
F_{\Pi.1} &: \mathsf{Tm}\,\big(F\,\Gamma \triangleright F\,(\Pi\,A\,B)\big)\,\big(\Pi\,(F\,A)\,(F\,B[F_{\triangleright.2}])\,[\mathsf{p}]\big) \\
F_{\mathsf{U}.1} &: \mathsf{Tm}\,(F\,\Gamma \triangleright F\,(\mathsf{U}\,i))\,(\mathsf{U}\,i) \\
F_{\mathsf{Bool}.2} &: \mathsf{Tm}\,(F\,\Gamma \triangleright \mathsf{Bool})\,(F\,\mathsf{Bool}[\mathsf{p}]) \\
F_{\Sigma} &: F\,(\Sigma_{\mathcal{S}}\,A\,B) \cong_{F\,\Gamma} \Sigma_{\mathcal{M}}\,(F\,A)\,(F\,B[F_{\triangleright.2}]) \\
F_{\top} &: F\,\top \cong_{F\,\Gamma} \top
\end{aligned}$$

The definition of $F_{\mathsf{U}.1}$ is similar to that of $F_{\Pi.1}$: we first use the eliminator El on a variable, then apply F, then adjust the context using $F_{\triangleright.2}$, finally use the constructor c. For Bool, we go in the other direction and use the eliminator in $\mathcal{M}$ on a variable and return the corresponding constructors in $\mathcal{S}$ with F applied to them. For Σ we combine these methods to obtain maps in both directions. The maps for $\top$ are trivial.

$$\begin{aligned}
F_{\Pi.1} &:= \mathsf{lam}\,\big(F\,(\mathsf{app}\,\mathsf{q})[F_{\triangleright.2} \circ (F_{\triangleright.2}{}^{\uparrow})]\big) \\
F_{\mathsf{U}.1} &:= \mathsf{c}\,\big(F\,(\mathsf{El}\,\mathsf{q})[F_{\triangleright.2}]\big) \\
F_{\mathsf{Bool}.2} &:= \mathsf{if}\,\_\,\mathsf{q}\,(F\,\mathsf{true}[\mathsf{p}])\,(F\,\mathsf{false}[\mathsf{p}]) \\
F_{\Sigma.1} &:= \big(F\,(\mathsf{projl}\,\mathsf{q}), F\,(\mathsf{projr}\,\mathsf{q})\big)[F_{\triangleright.2}] \\
F_{\Sigma.2} &:= F\,(1,0)\big[F_{\triangleright.2} \circ (F_{\triangleright.2} \circ (\mathsf{p}, \mathsf{projl}\,\mathsf{q}), \mathsf{projr}\,\mathsf{q})\big] \\
F_{\Sigma.12} &: \quad F_{\Sigma.1}[\mathsf{p}, F_{\Sigma.2}] = \\
&\quad\ (F\,(\mathsf{projl}\,\mathsf{q}), F\,(\mathsf{projr}\,\mathsf{q}))[F\,(\mathsf{p}, (1,0))][F_{\triangleright.2} \circ (F_{\triangleright.2} \circ (\mathsf{p}, \mathsf{projl}\,\mathsf{q}), \mathsf{projr}\,\mathsf{q})] = \\
&\quad\ (\mathsf{projl}\,\mathsf{q}, \mathsf{projr}\,\mathsf{q}) \overset{\Sigma\eta_{\mathcal{M}}}{=} \mathsf{q} \\
F_{\Sigma.21} &: \quad F_{\Sigma.2}[\mathsf{p}, F_{\Sigma.1}] = \\
&\quad\ F\,(1,0)[F\,(\mathsf{p}, \mathsf{projlq}, \mathsf{projr}\,q) \circ F_{\triangleright.2}] = \\
&\quad\ F\,(\mathsf{projl}\,\mathsf{q}, \mathsf{projr}\,q)[F_{\triangleright.2}] \overset{\Sigma\eta_{\mathcal{S}}}{=} F\,\mathsf{q}[F_{\triangleright.2}] \overset{F_{\mathsf{q}}, F_{\triangleright}}{=} \mathsf{q} \\
F_{\top.1} &:= \mathsf{tt} \\
F_{\top.2} &:= F\,\mathsf{tt}[\mathsf{p}] \\
F_{\top.12} &: \quad F_{\top.1}[\mathsf{p}, F_{\top.2}] = \mathsf{tt}[\mathsf{p}, F\,\mathsf{tt}[\mathsf{p}]] = \mathsf{tt} \overset{\top\eta_{\mathcal{M}}}{=} \mathsf{q} \\
F_{\top.21} &: \quad F_{\top.2}[\mathsf{p}, F_{\top.1}] = F\,\mathsf{tt}[\mathsf{p}] = F\,(\mathsf{tt}[\mathsf{p}])[F_{\triangleright.2}] \overset{\top\eta_{\mathcal{S}}}{=} F\,\mathsf{q}[F_{\triangleright.2}] = \mathsf{q}
\end{aligned}$$

At the end of Section 6 we remark on the (im)possibility of comparison maps in the other direction such as $F_{\Pi.2}$.

5 Gluing

In this section, given a pseudomorphism F from model $\mathcal{S}$ to model $\mathcal{M}$, we define a displayed model P^F (P for short) over $\mathcal{S}$. We call this model *gluing* along F and its components are given in Figure 3. We omit some ${}_{\mathcal{S}}$ and all ${}_{\mathcal{M}}$ subscripts for readability.

$$
\begin{aligned}
&\mathsf{Con}_{\mathsf{P}}\, i\, \Gamma && := \mathsf{Ty}\, i\, (F\, \Gamma)\\
&\mathsf{Ty}_{\mathsf{P}}\, j\, \Gamma_{\mathsf{P}}\, A && := \mathsf{Ty}\, j\, (F\, \Gamma \triangleright \Gamma_{\mathsf{P}} \triangleright F\, A[\mathsf{p}])\\
&\mathsf{Sub}_{\mathsf{P}}\, \Gamma_{\mathsf{P}}\, \Delta_{\mathsf{P}}\, \sigma && := \mathsf{Tm}\, (F\, \Gamma \triangleright \Gamma_{\mathsf{P}})\, (\Delta_{\mathsf{P}}[F\, \sigma \circ \mathsf{p}])\\
&\mathsf{Tm}_{\mathsf{P}}\, \Gamma_{\mathsf{P}}\, A_{\mathsf{P}}\, t && := \mathsf{Tm}\, (F\, \Gamma \triangleright \Gamma_{\mathsf{P}})\, (A_{\mathsf{P}}[\mathsf{id}, F\, t[\mathsf{p}]])\\
&\mathsf{id}_{\mathsf{P}} && := \mathsf{q}\\
&\sigma_{\mathsf{P}} \circ_{\mathsf{P}} \delta_{\mathsf{P}} && := \sigma_{\mathsf{P}}[F\, \delta \circ \mathsf{p}, \delta_{\mathsf{P}}]\\
&A_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}} && := A_{\mathsf{P}}[F\, \sigma \circ \mathsf{p}^2, \sigma_{\mathsf{P}}[\mathsf{p}], \mathsf{q}]\\
&t_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}} && := t_{\mathsf{P}}[F\, \sigma \circ \mathsf{p}, \sigma_{\mathsf{P}}]\\
&\cdot_{\mathsf{P}} && := \top\\
&\epsilon_{\mathsf{P}} && := \mathsf{tt}\\
&\Gamma_{\mathsf{P}} \triangleright_{\mathsf{P}} A_{\mathsf{P}} && := \Sigma\, \big(\Gamma_{\mathsf{P}}[\mathsf{p} \circ F_{\triangleright.1}]\big)\, \big(A_{\mathsf{P}}[\mathsf{p} \circ F_{\triangleright.1} \circ \mathsf{p}, 0, \mathsf{q}[F_{\triangleright.1} \circ \mathsf{p}]]\big)\\
&\sigma_{\mathsf{P}} ,_{\mathsf{P}} t_{\mathsf{P}} && := (\sigma_{\mathsf{P}}, t_{\mathsf{P}})\\
&\mathsf{p}_{\mathsf{P}} && := \mathsf{projl}\, \mathsf{q}\\
&\mathsf{q}_{\mathsf{P}} && := \mathsf{projr}\, \mathsf{q}\\
&\Pi_{\mathsf{P}}\, A_{\mathsf{P}}\, B_{\mathsf{P}} && := \Pi\, \big(F\, A[\mathsf{p}^2]\big)\, \Big(\Pi\, \big(A_{\mathsf{P}}[\mathsf{p}^2, \mathsf{q}]\big)\, \big(B_{\mathsf{P}}\big[F_{\triangleright.2} \circ (\mathsf{p}^4, 1), (3, 0), F_{\Pi.1}[\mathsf{p}^4, 2]\, \$\, 1\big]\big)\Big)\\
&\mathsf{lam}\, t_{\mathsf{P}} && := \mathsf{lam}\, \Big(\mathsf{lam}\, \big(t_{\mathsf{P}}\big[F_{\triangleright.2} \circ (\mathsf{p}^3, 1), (2, 0)\big]\big)\Big)\\
&\mathsf{app}\, t_{\mathsf{P}} && := \big(\mathsf{app}\, (\mathsf{app}\, t_{\mathsf{P}})\big)\big[\mathsf{p} \circ F_{\triangleright.1} \circ \mathsf{p}, \mathsf{projl}\, 0, 0[F_{\triangleright.1} \circ \mathsf{p}], \mathsf{projr}\, 0\big]\\
&\Sigma_{\mathsf{P}}\, A_{\mathsf{P}}\, B_{\mathsf{P}} && := \Sigma\, \big(A_{\mathsf{P}}\big[\mathsf{p}, \mathsf{projl}\, (F_{\Sigma.1}[\mathsf{p}^2, \mathsf{q}])\big]\big)\\
& && \qquad \big(B_{\mathsf{P}}\big[F_{\triangleright.2} \circ (\mathsf{p}^3, \mathsf{projl}\, (F_{\Sigma.1}[\mathsf{p}^2, \mathsf{q}])), (2, 0), \mathsf{projr}\, (F_{\Sigma.1}[\mathsf{p}^2, \mathsf{q}])\big]\big)\\
&(u_{\mathsf{P}} ,_{\mathsf{P}} v_{\mathsf{P}}) && := (u_{\mathsf{P}}, v_{\mathsf{P}})\\
&\mathsf{projl}_{\mathsf{P}}\, t_{\mathsf{P}} && := \mathsf{projl}\, t_{\mathsf{P}}\\
&\mathsf{projr}_{\mathsf{P}}\, t_{\mathsf{P}} && := \mathsf{projr}\, t_{\mathsf{P}}\\
&\top_{\mathsf{P}} && := \top\\
&\mathsf{tt}_{\mathsf{P}} && := \mathsf{tt}\\
&\mathsf{U}_{\mathsf{P}}\, i && := \mathsf{El}\, (F_{\mathsf{U}.1}[\mathsf{p}^2, \mathsf{q}]) \Rightarrow \mathsf{U}\, i\\
&\mathsf{El}_{\mathsf{P}}\, a_{\mathsf{P}} && := \mathsf{El}\, (\mathsf{app}\, a_{\mathsf{P}})\\
&\mathsf{c}_{\mathsf{P}}\, A_{\mathsf{P}} && := \mathsf{lam}\, (\mathsf{c}\, A_{\mathsf{P}})\\
&\mathsf{Bool}_{\mathsf{P}} && := \Sigma\, \mathsf{Bool}\, \big(\mathsf{Id}\, (F\, \mathsf{Bool}_{\mathcal{S}}[\mathsf{p}^3])\, (F_{\mathsf{Bool}.2}[\mathsf{p}^3, \mathsf{q}])\, 1\big)\\
&\mathsf{true}_{\mathsf{P}} && := (\mathsf{true}, \mathsf{refl}\, (F\, \mathsf{true}_{\mathcal{S}}[\mathsf{p}]))\\
&\mathsf{false}_{\mathsf{P}} && := (\mathsf{false}, \mathsf{refl}\, (F\, \mathsf{false}_{\mathcal{S}}[\mathsf{p}]))\\
&\mathsf{if}_{\mathsf{P}}\, C_{\mathsf{P}}\, t_{\mathsf{P}}\, u_{\mathsf{P}}\, v_{\mathsf{P}} && := \mathsf{J}\, _\, (\mathsf{if}\, _\, (\mathsf{projl}\, t_{\mathsf{P}})\, u_{\mathsf{P}}\, v_{\mathsf{P}})\, (\mathsf{projr}\, t_{\mathsf{P}})
\end{aligned}
$$

Figure 3 The displayed model P^F obtained by gluing along F. We write P instead of P^F, we omit some $_{\mathcal{S}}$ and all $_{\mathcal{M}}$ subscripts for readability. The full version of if_{P} (with the _s filled in) is given in Appendix B.

In the introduction we remarked that in categorical gluing an object in the glued model consists of a triple $\Gamma : |\mathcal{S}|$, $\Delta : |\mathcal{M}|$ and a morphism $\mathcal{M}(\Delta, F\,\Gamma)$. We could follow this line and define the gluing as a model with contexts such triples that comes with a strict "projection" morphism to $\mathcal{S}$. This could be called the fibrational or display map approach. Instead our definition is more type theoretic, it uses indexed families, doubly (for the correspondence between fibrations and families see e.g. [8, p. 221]). Firstly, the glued model is given as a displayed model, that is, for each $\Gamma : \mathsf{Con}_{\mathcal{S}}\,i$ we have a set $\mathsf{Con}_{\mathsf{P}}\,i\,\Gamma$. Secondly, instead of setting $\mathsf{Con}_{\mathsf{P}}\,i\,\Gamma$ to $(\Delta : \mathsf{Con}_{\mathcal{M}}\,i) \times \mathsf{Sub}_{\mathcal{M}}\,\Delta\,(F\,\Gamma)$, we use the built-in notion of indexed families in $\mathcal{M}$, that is: types. Hence a context over Γ is an $\mathcal{M}$-type in context $F\,\Gamma$. We remark that the glueing construction also works with the former choice of contexts.

Types in type theory can be thought of as proof-relevant predicates over their context and this is the intuition we adopt for describing the glued model. This is in line with the logical predicate view of gluing. We start with $\mathsf{Con}_{\mathsf{P}}\,i\,\Gamma$: a predicate at Γ is indexed over the F-image of Γ. A predicate at a type A is indexed over the image of Γ for which the predicate holds and the image of A. For a substitution $\sigma : \mathsf{Sub}\,\Gamma\,\Delta$, we state the fundamental lemma: if the predicate holds at Γ, the predicate holds at Δ for the F-image of the substitution. In short, images of substitutions respect the predicate. For terms, we similarly state that the image of a term respects the predicate.

We continue by explaining what the logical predicate says at different contexts and types. The predicate at the empty context $\cdot_{\mathsf{P}}$ is always true. At extended contexts the predicate is given pointwise by a Σ-type. $\Gamma_{\mathsf{P}} \rhd A_{\mathsf{P}}$ is in context $F\,(\Gamma \rhd A)$, but Γ_{P} only needs the component $F\,\Gamma$, which we obtain using the isomorphism $F_{\rhd.1}$ from $F\,(\Gamma \rhd_{\mathcal{S}} A)$ to $F\,\Gamma \rhd F\,A$ followed by first projection. A_{P} is first indexed over $F\,\Gamma$, which is given by $\mathsf{p} \circ F_{\rhd.1} \circ \mathsf{p}$, then over Γ_{P}, which is the first component of the Σ-type referenced by q, then over $F\,A$, which is provided by the $F_{\rhd.1}$ part of the isomorphism.

The predicate at a Π-type holds for a function of type $F\,(\Pi\,A\,B)$ if whenever it holds for an input, it holds for the output. Let's look at how we express that the predicate holds at B for the output! We are in context

$$\Theta := F\,\Gamma \rhd \underbrace{\Gamma_{\mathsf{P}}}_{3} \rhd \underbrace{F\,(\Pi_{\mathcal{S}}\,A\,B)}_{2} \rhd \underbrace{F\,A[\mathsf{p}^2]}_{1} \rhd \underbrace{A_{\mathsf{P}}[\mathsf{p}^2, \mathsf{q}]}_{0}$$

where we wrote the de Bruijn indices referring to each component underneath. B_{P} is a predicate indexed over $F\,(\Gamma \rhd_{\mathcal{S}} A)$, $\Gamma_{\mathsf{P}} \rhd_{\mathsf{P}} A_{\mathsf{P}}$ and $F\,B[\mathsf{p}]$. The first index is given by $F_{\rhd.2}$, which puts together the $F\,\Gamma$ (forgetting the last four elements in Θ by p^4) and the $F\,A$ components (last but one element in Θ, i.e. 1). The second index is given by de Bruijn indices 3 and 0. The last index is the result of applying the function given by De Bruijn index 2. We have to use the comparison map $F_{\Pi.1}$ defined in Section 4 which turns an $F\,(\Pi\,A\,B)$ into a $\Pi\,(F\,A)\,(F\,B[F_{\rhd.2}])$. We supply its dependencies $F\,\Gamma$ by p^4 and $F\,(\Pi\,A\,B)$ by 2 and we use old-style application \$ with input 1 to get the result.

The predicate at a Σ-type holds if it holds pointwise. Here we use the comparison map $F_{\Sigma.1}$ combined with projl and projr to obtain $F\,A$ and $F\,B$ from $F\,(\Sigma_{\mathcal{S}}\,A\,B)$. The predicate at $\top$ is trivial. The predicate at the universe is the space of predicates expressed as functions into $\mathsf{U}\,i$. The domain of this function space is again obtained by applying the comparison map $F_{\mathsf{U}.1}$. The predicate at Bool for b in $F\,\mathsf{Bool}$ says that there is an $\mathcal{M}$-boolean to which we apply the comparison map $F_{\mathsf{Bool}.2}$ (which turns it into $F\,\mathsf{Bool}$), the result is equal to b.

The substition and term part of the gluing model is fairly straightforward. The most interesting component is if_{P} where we use J to eliminate the right projection of t_{P} (which is the equality in the second component of $\mathsf{Bool}_{\mathsf{P}}$), then we case split on the first projection by $\mathsf{if}_{\mathcal{M}}$ and return u_{P} and v_{P} in the true and false cases, respectively. We omitted some arguments of J and if for readability, the full version is given in Appendix B. There we also verify that all equalities of the displayed model of type theory hold.

6 Global section functor

In this section we define the global section functor and show that it is a pseudomorphism. In the next section we will use this property to derive canonicity for type theory.

A model $\mathcal{S}$ *supports a global section functor* if it has the following two properties:

- $\mathsf{Sub}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ \Gamma : \mathsf{Set}_i$ whenever $\Gamma : \mathsf{Con}_{\mathcal{S}}\ i$
- $\mathsf{Tm}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ (A[\rho]_{\mathcal{S}}) : \mathsf{Set}_j$ whenever $A : \mathsf{Ty}_{\mathcal{S}}\ j\,\Gamma$, $\rho : \mathsf{Sub}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ \Gamma$.

For example, the syntax S (defined at the end of Section 2) supports a global section functor because syntactic substitutions and terms are in the lowest metatheoretic universe and this universe hierarchy is cumulative. The Set model (defined in Section 3) also supports a global section functor because $\Gamma : \mathsf{Con}_{\mathsf{Set}}\ i$ means $\Gamma : \mathsf{Set}_i$ and $\mathsf{Sub}_{\mathsf{Set}}\ \cdot_{\mathsf{Set}}\ \Gamma = \mathbb{1} \to \Gamma : \mathsf{Set}_i$, and similarly for the second condition.

The *global section functor* $\mathsf{G}^{\mathcal{S}}$ is a pseudomorphism from such an $\mathcal{S}$ to the set model Set of Section 3. It maps a context to the set of closed substitutions into that context. It maps a type to the function sending a closed substitution to the set of closed terms of the type substituted by the input substitution. Substitutions and terms are mapped to postcomposition and substitution by the closed substitution, respectively. We write G instead of $\mathsf{G}^{\mathcal{S}}$ for readability.

$$\begin{aligned}
\mathsf{G}_{\mathsf{Con}}\,\Gamma &: \underbrace{\mathsf{Con}_{\mathsf{Set}}\,i}_{=\mathsf{Set}_i} && := \mathsf{Sub}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ \Gamma\\
\mathsf{G}_{\mathsf{Ty}}\,A &: \underbrace{\mathsf{Ty}_{\mathsf{Set}}\,j\,(\mathsf{G}\,\Gamma)}_{=\mathsf{G}\,\Gamma\to\mathsf{Set}_j} && := \lambda\rho\,.\,\mathsf{Tm}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ (A[\rho]_{\mathcal{S}})\\
\mathsf{G}_{\mathsf{Sub}}\,\sigma &: \underbrace{\mathsf{Sub}_{\mathsf{Set}}\,(\mathsf{G}\,\Gamma)\,(\mathsf{G}\,\Delta)}_{=\mathsf{G}\,\Gamma\to\mathsf{G}\,\Delta} && := \lambda\rho\,.\,\sigma\circ_{\mathcal{S}}\rho\\
\mathsf{G}_{\mathsf{Tm}}\,t &: \underbrace{\mathsf{Tm}_{\mathsf{Set}}\,(\mathsf{G}\,\Gamma)\,(\mathsf{G}\,A)}_{=(\rho:\mathsf{G}\,\Gamma)\to\mathsf{G}\,A\,\rho} && := \lambda\rho\,.\,t[\rho]_{\mathcal{S}}
\end{aligned}$$

Note that this pseudomorphism is indeed weak on the empty context: $\mathsf{Sub}_{\mathcal{S}}\ \cdot\ \cdot$ is isomorphic to $\mathbb{1}$ (the empty context in Set) by $\cdot\eta_{\mathcal{S}}$, but not necessarily equal. Similarly, $\mathsf{Sub}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ (\Gamma \triangleright_{\mathcal{S}} A)$ is isomorphic to $(\rho : \mathsf{Sub}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ \Gamma) \times \mathsf{Tm}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ (A[\rho]_{\mathcal{S}})$ by comprehension ($\triangleright\beta_{1\mathcal{S}}$, $\triangleright\beta_{2\mathcal{S}}$, $\triangleright\eta_{\mathcal{S}}$), but not necessarily equal. In Appendix A we show that G is indeed a pseudomorphism satisfying the conditions in Figure 2.

Remark on comparison maps. In Section 4 we showed that pseudomorphisms support certain comparison maps, for example

$$F_{\Pi.1} : \mathsf{Tm}\,\big(F\,\Gamma \triangleright F\,(\Pi\,A\,B)\big)\,\big(\Pi\,(F\,A)\,(F\,B[F_{\triangleright.2}])\,[\mathsf{p}]\big)$$

can be defined for any pseudomorphism F. The global section functor gives a good way to show that this comparison map is not an isomorphism in general (as opposed to $F_{\Sigma.1}$). The other direction would be a

$$\mathsf{G}_{\Pi.2} : \underbrace{\mathsf{Tm}_{\mathsf{Set}}\,\big(\mathsf{G}\,\Gamma \triangleright_{\mathsf{Set}} \Pi_{\mathsf{Set}}\,(\mathsf{G}\,A)\,(\mathsf{G}\,B[\mathsf{G}_{\triangleright.2}])\big)\,\big(\mathsf{G}\,(\Pi_{\mathcal{S}}\,A\,B)[\mathsf{p}]\big)}_{=(\rho:\mathsf{Sub}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ \Gamma)\times\big((u:\mathsf{Tm}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ (A[\rho]_{\mathcal{S}}))\to\mathsf{Tm}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ (B[\rho,u]_{\mathcal{S}})\big)\to\mathsf{Tm}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ (\Pi_{\mathcal{S}}\,A\,B[\rho]_{\mathcal{S}})},$$

providing a way to turn a metatheoretic function between terms into a term of a function type in $\mathcal{S}$. However if e.g. $S = \mathsf{S}$, $A = \mathsf{Nat}$ and $B = \mathsf{Bool}$, then following Cantor there are more metatheoretic functions from natural numbers to booleans than terms. Similarly, if $\mathsf{G}_{\mathsf{Bool}.2}$ was an isomorphism, it would have an inverse

$$\mathsf{G}_{\mathsf{Bool}.1} : \underbrace{\mathsf{Tm}_{\mathsf{Set}}\,(\mathsf{G}\,\Gamma \triangleright_{\mathsf{Set}} \mathsf{G}\,\mathsf{Bool}_{\mathcal{S}})\,\mathsf{Bool}_{\mathsf{Set}}}_{=(\rho:\mathsf{Sub}_{\mathcal{S}}\,\cdot_{\mathcal{S}}\,\Gamma)\times\mathsf{Tm}_{\mathcal{S}}\,\cdot_{\mathcal{S}}\,\mathsf{Bool}_{\mathcal{S}}\to\mathbb{2}},$$

but if $\mathcal{S}$ is a model without equality reflection where booleans are defined by a quotient then there are more than two terms of type Bool (while internally to $\mathcal{S}$ there are only two elements).

7 Reaping the fruits

Let I be the identity morphism from S to S, which is obviously a strict morphism, hence pseudo.[2] Elimination into gluing along I produces a function whose input is a term t in context Γ and whose output is a term in context Γ extended by $\mathsf{elim}^{\mathsf{P}^{\mathsf{I}}}_{\mathsf{Con}}\,\Gamma$ which expresses that the predicate holds at Γ. The type of the output term says that the predicate holds at A for t. This is the fundamental lemma or parametricity theorem.

$$\mathsf{elim}^{\mathsf{P}^{\mathsf{I}}}_{\mathsf{Tm}} : (t : \mathsf{Tm}_{\mathsf{S}}\,\Gamma\,A) \to \mathsf{Tm}_{\mathsf{S}}\,(\Gamma \triangleright \mathsf{elim}^{\mathsf{P}^{\mathsf{I}}}_{\mathsf{Con}}\,\Gamma)\,\big((\mathsf{elim}^{\mathsf{P}^{\mathsf{I}}}_{\mathsf{Ty}}\,A)[\mathsf{id}, t[\mathsf{p}]]\big)$$

Let us look at the "hello world" example of parametricity, the case where $\Gamma = \cdot$ and $A = \Pi\,(\mathsf{U}\,i)\,(\mathsf{El}\,\mathsf{q} \Rightarrow \mathsf{El}\,\mathsf{q})$. Now using the fact that $\mathsf{elim}^{\mathsf{P}^{\mathsf{I}}}$ is a section, the type of $\mathsf{elim}^{\mathsf{P}^{\mathsf{I}}}_{\mathsf{Tm}}\,t$ computes to

$$\mathsf{Tm}_{\mathsf{S}}\,(\cdot \triangleright \top)\,\Big(\Pi\,(\mathsf{U}\,i)\,\Big(\Pi\,(\mathsf{El}\,\mathsf{q} \Rightarrow \mathsf{U}\,i)\,\Big(\Pi\,(\mathsf{El}\,1)\,\big(\mathsf{El}\,(1\,\$\,0) \Rightarrow \mathsf{El}\,(1\,\$(t\,\$\,2\,\$\,0))\big)\Big)\Big)\Big),$$

where the type is the object theoretic syntax for

$$(A : \mathsf{Set}_i)(C : A \to \mathsf{Set}_i)(a : A) \to C\,a \to C\,(t\,A\,a).$$

Given a fixed type $A : \mathsf{Ty}_{\mathsf{S}}\,i\,\cdot$ and an element $u : \mathsf{Tm}_{\mathsf{S}}\,\cdot\,A$ we have

$$(\mathsf{elim}^{\mathsf{P}^{\mathsf{I}}}_{\mathsf{Tm}}\,t)[\epsilon, \mathsf{tt}]\,\$\,\mathsf{c}\,A\,\$\,\mathsf{lam}\,(\mathsf{c}\,(\mathsf{Id}\,(A[\epsilon])\,0\,u[\epsilon]))\,\$\,u\,\$\,\mathsf{refl}\,u : \mathsf{Tm}\,\cdot\,\big(\mathsf{Id}\,A\,(t\,\$\,c\,A\,\$\,u)\,u\big),$$

that is, we get that for any A and u, $t\,\$\,\mathsf{c}\,A\,\$\,u$ is equal to u.

Eliminating into gluing along $\mathsf{rec}^{\mathsf{Set}}$ (the interpretation into the set model, see end of Section 3) produces Reynolds-style parametricity. It says that if there is an interpretation of the context Γ for which the predicate holds at Γ, the predicate holds at A for the interpretation of t.

$$\mathsf{elim}^{\mathsf{P}^{\mathsf{rec}^{\mathsf{Set}}}}_{\mathsf{Tm}} : (t : \mathsf{Tm}_{\mathsf{S}}\,\Gamma\,A) \to (\gamma : [\![\Gamma]\!]) \times (\bar{\gamma} : \mathsf{elim}^{\mathsf{P}^{\mathsf{rec}^{\mathsf{Set}}}}_{\mathsf{Con}}\,\Gamma\,\gamma) \to \mathsf{elim}^{\mathsf{P}^{\mathsf{rec}^{\mathsf{Set}}}}_{\mathsf{Ty}}\,A\,(\gamma, \bar{\gamma}, [\![t]\!]\,\gamma)$$

Eliminating into gluing along the global section functor G^{S} from the syntax to the set model gives the following.

$$\mathsf{elim}^{\mathsf{P}^{\mathsf{G}}}_{\mathsf{Tm}} : (t : \mathsf{Tm}_{\mathsf{S}}\,\Gamma\,A) \to (\rho : \mathsf{Sub}_{\mathsf{S}}\,\cdot\,\Gamma) \times (\bar{\rho} : \mathsf{elim}^{\mathsf{P}^{\mathsf{G}}}_{\mathsf{Con}}\,\Gamma\,\rho) \to \mathsf{elim}^{\mathsf{P}^{\mathsf{G}}}_{\mathsf{Ty}}\,A\,(\rho, \bar{\rho}, t[\rho])$$

If t is a boolean in the empty context, the type of $\mathsf{elim}^{\mathsf{P}^{\mathsf{G}}}_{\mathsf{Tm}}\,t\,(\mathsf{id}, *)$ is $\mathsf{elim}^{\mathsf{P}^{\mathsf{G}}}_{\mathsf{Ty}}\,\mathsf{Bool}\,(\rho, *, t)$ which is equal to $(b : \mathbb{2}) \times (\mathsf{case}\,b\,\mathsf{true}_{\mathsf{S}}\,\mathsf{false}_{\mathsf{S}} = t)$, i.e. canonicity.

[2] Note that the target of the pseudomorphism needs to have identity types, so technically I is the embedding of the syntax without identity types into the syntax with identity types. Alternatively, we can extend gluing for identity types.

8 Conclusions and further work

In this paper we defined gluing for pseudomorphisms of models of type theory thus generalising parametricity and canonicity. We did not try to derive the most general notion of gluing, e.g. we require that the target model supports $\top$-, Σ-, Id-types in addition to what we have in the domain model. It would have been possible to give a less indexed variant of gluing where $\top$ and Σ are not needed, but Id types (or (F Bool)-indexed inductive families) would be still required to support gluing for Bool. A less indexed variant however would be more tedious to work with because the glued model would involve some metatheoretic equalities.

In the future we would like to generalise our construction to richer type theories having an identity type, inductive and coinductive types. We believe that this is possible without any extra conditions.

Normalisation by evaluation (NBE) for type theory is also defined using a proof-relevant logical predicate [4]. This logical predicate is given by gluing along the Yoneda embedding from the syntax to the presheaf model over the category of contexts and renamings. This is a pseudomorphism, so we obtain a glued model using our method. However, the universe in this model is not what we want. As a second step after gluing, NBE requires the definition of quote and unquote (sometimes called reify and reflect) functions from terms for which the predicate holds to normal forms and from neutral terms to witnesses of the predicate, respectively. We need to include these as part of the universe in the glued model to make the construction work. The predicate for Bool also needs to be adjusted.

We would also like to investigate examples of non-strict pseudomorphisms apart from global section and Yoneda for which the construction in this paper could be useful; for example, to derive canonicity proofs for type theories justified by models other than the set model.

References

1 Andreas Abel, Joakim Öhman, and Andrea Vezzosi. Decidability of conversion for type theory in type theory. *PACMPL*, 2:23:1–23:29, 2017.

2 Thorsten Altenkirch, Martin Hofmann, and Thomas Streicher. Reduction-free normalisation for system F. Unpublished draft, 1997.

3 Thorsten Altenkirch and Ambrus Kaposi. Type theory in type theory using quotient inductive types. In Rastislav Bodik and Rupak Majumdar, editors, *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2016, St. Petersburg, FL, USA, January 20 - 22, 2016*, pages 18–29. ACM, 2016. doi:10.1145/2837614.2837638.

4 Thorsten Altenkirch and Ambrus Kaposi. Normalisation by Evaluation for Type Theory, in Type Theory. *Logical Methods in Computer Science*, Volume 13, Issue 4, October 2017. doi:10.23638/LMCS-13(4:1)2017.

5 Michael Artin, Alexander Grothendieck, and Jean-Louis Verdier. *Theorie de Topos et Cohomologie Etale des Schemas I*, volume 269 of *Lecture Notes in Mathematics*. Springer, 1971.

6 Robert Atkey, Neil Ghani, and Patricia Johann. A relationally parametric model of dependent type theory. In Suresh Jagannathan and Peter Sewell, editors, *The 41st Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL '14, San Diego, CA, USA, January 20-21, 2014*, pages 503–516. ACM, 2014. doi:10.1145/2535838.2535852.

7 Jean-Philippe Bernardy, Patrik Jansson, and Ross Paterson. Proofs for Free — Parametricity for Dependent Types. *Journal of Functional Programming*, 22(02):107–152, 2012. doi:10.1017/S0956796812000056.

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

9 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.

10 Thierry Coquand. Canonicity and normalisation for Dependent Type Theory. *CoRR*, abs/1810.09367, 2018. arXiv:1810.09367.

11 Roy L. Crole. *Categories for types.* Cambridge mathematical textbooks. Cambridge University Press, Cambridge, New York, 1993. URL: http://opac.inria.fr/record=b1088776.

12 Peter Dybjer. Internal Type Theory. In *Lecture Notes in Computer Science*, pages 120–134. Springer, 1996.

13 Marcelo Fiore and Alex Simpson. Lambda Definability with Sums via Grothendieck Logical Relations. In Jean-Yves Girard, editor, *Typed Lambda Calculi and Applications*, pages 147–161, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

14 Marcelo P. Fiore. Semantic analysis of normalisation by evaluation for typed lambda calculus. In *Proceedings of the 4th international ACM SIGPLAN conference on Principles and practice of declarative programming, October 6-8, 2002, Pittsburgh, PA, USA (Affiliated with PLI 2002)*, pages 26–37. ACM, 2002. doi:10.1145/571157.571161.

15 Claudio Hermida, Uday S. Reddy, and Edmund P. Robinson. Logical Relations and Parametricity — A Reynolds Programme for Category Theory and Programming Languages. *Electronic Notes in Theoretical Computer Science*, 303(0):149–180, 2014. Proceedings of the Workshop on Algebra, Coalgebra and Topology (WACT 2013). doi:10.1016/j.entcs.2014.02.008.

16 Martin Hofmann. Conservativity of Equality Reflection over Intensional Type Theory. In *TYPES 95*, pages 153–164, 1995.

17 B. Jacobs. *Categorical Logic and Type Theory.* Number 141 in Studies in Logic and the Foundations of Mathematics. North Holland, Amsterdam, 1999.

18 Ambrus Kaposi, András Kovács, and Thorsten Altenkirch. Constructing Quotient Inductive-inductive Types. *Proc. ACM Program. Lang.*, 3(POPL):2:1–2:24, January 2019. doi:10.1145/3290315.

19 Chung kil Hur and Derek Dreyer. A Kripke logical relation between ML and assembly. *Conference Record of the Annual ACM Symposium on Principles of Programming Languages*, pages 133–146, January 2010. doi:10.1145/1926385.1926402.

20 András Kovács. Formalisation of canonicity for type theory in Agda, November 2018. URL: https://github.com/AndrasKovacs/glue.

21 J. Lambek and P. J. Scott. *Introduction to higher order categorical logic.* Cambridge University Press, New York, NY, USA, 1986.

22 Gordon D. Plotkin. Lambda-Definability and Logical Relations. Memorandum SAI–RM–4, University of Edinburgh, Edinburgh, Scotland, October 1973.

23 Florian Rabe and Kristina Sojakova. Logical Relations for a Logical Framework. *ACM Trans. Comput. Logic*, 14(4):32:1–32:34, November 2013. doi:10.1145/2536740.2536741.

24 John C. Reynolds. Types, Abstraction and Parametric Polymorphism. In R. E. A. Mason, editor, *Information Processing 83, Proceedings of the IFIP 9th World Computer Congress, Paris, September 19-23, 1983*, pages 513–523. Elsevier Science Publishers B. V. (North-Holland), Amsterdam, 1983.

25 Michael Shulman. Univalence for inverse diagrams and homotopy canonicity. *Mathematical Structures in Computer Science*, 25:1203–1277, June 2015. arXiv:1203.3253. doi:10.1017/S0960129514000565.

26 Jonathan Sterling and Bas Spitters. Normalization by gluing for free λ-theories. *CoRR*, abs/1809.08646, 2018. arXiv:1809.08646.

27 Andrew W. Appel and David Mcallester. An indexed model of recursive types for foundational proof-carrying code. *ACM Trans. Program. Lang. Syst.*, 23:657–683, September 2001. doi:10.1145/504709.504712.

A The global section functor is a pseudomorphism

Here we verify the equalities of a pseudomorphism (see Section 4) for the global section functor from a model $\mathcal{S}$ to Set (see Section 3). The definition of the global section functor is given in Section 6, here we repeat it to save turning pages.

$$\begin{array}{ll}
\mathsf{G}_{\mathsf{Con}}\,\Gamma & := \mathsf{Sub}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ \Gamma \\
\mathsf{G}_{\mathsf{Ty}}\,A & := \lambda\rho\,.\,\mathsf{Tm}_{\mathcal{S}}\ \cdot_{\mathcal{S}}\ (A[\rho]_{\mathcal{S}}) \\
\mathsf{G}_{\mathsf{Sub}}\,\sigma & := \lambda\rho\,.\,\sigma\circ_{\mathcal{S}}\rho \\
\mathsf{G}_{\mathsf{Tm}}\,t & := \lambda\rho\,.\,t[\rho]_{\mathcal{S}}
\end{array}$$

$$\begin{array}{lll}
\mathsf{G}_{\mathsf{id}} & : & \mathsf{G}\,\mathsf{id}_{\mathcal{S}} = \lambda\rho.\mathsf{id}\circ_{\mathcal{S}}\rho \overset{\mathsf{idl}_{\mathcal{S}}}{=} \lambda\rho.\rho = \mathsf{id}_{\mathsf{Set}} \\
\mathsf{G}_{\circ} & : & \mathsf{G}\,(\sigma\circ_{\mathcal{S}}\delta) = \lambda\rho.(\sigma\circ_{\mathcal{S}}\delta)\circ_{\mathcal{S}}\rho \overset{\mathsf{ass}_{\mathcal{S}}}{=} \lambda\rho.\sigma\circ_{\mathcal{S}}(\delta\circ_{\mathcal{S}}\rho) = \mathsf{G}\,\sigma\circ_{\mathsf{Set}}\mathsf{G}\,\delta \\
\mathsf{G}_{[]} & : & \mathsf{G}\,(A[\delta]_{\mathcal{S}}) = \lambda\rho.\mathsf{Tm}_{\mathcal{S}}\ \cdot\ (A[\delta][\rho]) \overset{[\circ]_{\mathcal{S}}}{=} \lambda\rho.\mathsf{Tm}_{\mathcal{S}}\ \cdot\ (A[\delta\circ\rho]) = (\mathsf{G}\,A)[\mathsf{G}\,\sigma]_{\mathsf{Set}} \\
\mathsf{G}_{[]} & : & \mathsf{G}\,(t[\delta]_{\mathcal{S}}) = \lambda\rho.t[\delta][\rho] \overset{[\circ]_{\mathcal{S}}}{=} \lambda\rho.t[\delta\circ\rho] = (\mathsf{G}\,t)[\mathsf{G}\,\sigma]_{\mathsf{Set}} \\
\mathsf{G}_{\cdot} & : & \mathsf{G}\cdot_{\mathcal{S}}\cong\cdot_{\mathsf{Set}} := (\lambda\rho.*, \lambda\_.\epsilon, \mathrm{trivial}, \mathrm{trivial}) \\
\mathsf{G}_{\epsilon} & : & \mathsf{G}\,\epsilon_{\mathcal{S}} = \lambda\rho.\epsilon\circ\rho \overset{\cdot\eta_{\mathcal{S}}}{=} \lambda\rho.\epsilon = \mathsf{G}_{\cdot.2}\circ_{\mathsf{Set}}\epsilon_{\mathsf{Set}} \\
\mathsf{G}_{\triangleright.1} & : & \mathsf{Sub}_{\mathsf{Set}}\,(\mathsf{G}\,(\Gamma\triangleright_{\mathcal{S}}A))\,(\mathsf{G}\,\Gamma\triangleright_{\mathsf{Set}}\mathsf{G}\,A) := \lambda\rho.(\mathsf{p}\circ\rho, \mathsf{q}[\rho]) \\
\mathsf{G}_{\triangleright.2} & : & \mathsf{Sub}_{\mathsf{Set}}\,(\mathsf{G}\,\Gamma\triangleright_{\mathsf{Set}}\mathsf{G}\,A)\,(\mathsf{G}\,(\Gamma\triangleright_{\mathcal{S}}A)) := \lambda(\rho,u).(\rho,_{\mathcal{S}}u) \\
\mathsf{G}_{\triangleright.12} & : & \mathsf{G}_{\triangleright.1}\circ_{\mathsf{Set}}\mathsf{G}_{\triangleright.2} = \lambda(\rho,u).(\mathsf{p}\circ(\rho,_{\mathcal{S}}u),_{\mathcal{S}}\mathsf{q}[\rho,_{\mathcal{S}},u]) \overset{\triangleright\beta_1,\triangleright\beta_2}{=} \lambda(\rho,u).(\rho,u) = \mathsf{id}_{\mathsf{Set}} \\
\mathsf{G}_{\triangleright.21} & : & \mathsf{G}_{\triangleright.2}\circ_{\mathsf{Set}}\mathsf{G}_{\triangleright.1} = \lambda\rho.(\mathsf{p}\circ\rho,_{\mathcal{S}}\mathsf{q}[\rho]) \overset{,\circ_{\mathcal{S}}}{=} \lambda\rho.(\mathsf{p},\mathsf{q})\circ\rho \overset{\triangleright\eta}{=} \lambda\rho.\mathsf{id}\circ\rho \overset{\mathsf{idl}_{\mathcal{S}}}{=} \lambda\rho.\rho = \mathsf{id}_{\mathsf{Set}} \\
\mathsf{G}_{\triangleright.1\circ} & : & \mathsf{G}_{\triangleright.1}\circ_{\mathsf{Set}}\ \mathsf{G}\,(\sigma^{\uparrow_{\mathcal{S}}}) = \lambda\rho.(\sigma\circ\mathsf{p}\circ\rho, \mathsf{q}[\rho]) \overset{,\circ_{\mathcal{S}}}{=} \lambda\rho.(\sigma\circ\mathsf{p},\mathsf{q})\circ\rho \overset{\mathsf{idl}_{\mathcal{S}},\triangleright\eta_{\mathcal{S}}}{=} \\
 & & \lambda\rho.(\sigma\circ\mathsf{p},\mathsf{q})\circ(\mathsf{p}\circ\rho,\mathsf{q}[\rho]) = (\mathsf{G}\,\sigma)^{\uparrow_{\mathsf{Set}}}\circ_{\mathsf{Set}}\mathsf{G}_{\triangleright.1} \\
\mathsf{G}_{,} & : & \mathsf{G}\,(\sigma,_{\mathcal{S}}t) = \lambda\rho.(\sigma,t)\circ\rho \overset{,\circ_{\mathcal{S}}}{=} \rho.(\sigma\circ\rho, t[\rho]) = \mathsf{G}_{\triangleright.2}\circ_{\mathsf{Set}}(\mathsf{G}\,\sigma,_{\mathsf{Set}}\mathsf{G}\,t) \\
\mathsf{G}_{\mathsf{p}} & : & \mathsf{G}\,\mathsf{p}_{\mathcal{S}} = \lambda\rho.\mathsf{p}\circ\rho = \lambda\rho.(\mathsf{p}\circ\rho,\mathsf{q}[\rho]).{}_1 = \mathsf{p}_{\mathsf{Set}}\circ_{\mathsf{Set}}\mathsf{G}_{\triangleright.1} \\
\mathsf{G}_{\mathsf{q}} & : & \mathsf{G}\,\mathsf{q}_{\mathcal{S}} = \lambda\rho.\mathsf{q}[\rho] = \lambda\rho.(\mathsf{p}\circ\rho,\mathsf{q}[\rho]).{}_2 = \mathsf{q}_{\mathsf{Set}}[\mathsf{G}_{\triangleright.1}]_{\mathsf{Set}}
\end{array}$$

B Full version of if_{P} and equalities in gluing

if_{P} is part of the glued displayed model P, see Section 5, Figure 3. Its definition is the following including the omitted _ arguments.

$$\begin{array}{l}
\mathsf{if}_{\mathsf{P}}\,C_{\mathsf{P}}\,t_{\mathsf{P}}\,u_{\mathsf{P}}\,v_{\mathsf{P}} := \\
\quad \mathsf{J}\,\Big(C_{\mathsf{P}}\big[F_{\triangleright.2}\circ(\mathsf{p}^3,1),(2,(\mathsf{projl}\,t_{\mathsf{P}}[\mathsf{p}^2],0)),F\,\big(\mathsf{if}_{\mathcal{S}}\,(C[\mathsf{p}^2,\mathsf{q}])\,\mathsf{q}\,(u[\mathsf{p}])\,(v[\mathsf{p}])\big)[F_{\triangleright.2}\circ(\mathsf{p}^3,1)]\big]\Big) \\
\qquad \Big(\mathsf{if}\,\big(C_{\mathsf{P}}\big[F_{\triangleright.2}\circ(\mathsf{p}^2,w),(1,(0,\mathsf{refl}\,w)),F\,\big(\mathsf{if}_{\mathcal{S}}\,(C[\mathsf{p}^2,\mathsf{q}])\,\mathsf{q}\,(u[\mathsf{p}])\,(v[\mathsf{p}])\big)[F_{\triangleright.2}\circ(\mathsf{p}^2,w)]\big]\big) \\
\qquad\quad (\mathsf{projl}\,t_{\mathsf{P}})\,u_{\mathsf{P}}\,v_{\mathsf{P}}\,\Big) \\
\qquad (\mathsf{projr}\,t_{\mathsf{P}})
\end{array}$$

where w abbreviates $\mathsf{if}\,(F\,\mathsf{Bool}_{\mathcal{S}}[\mathsf{p}^2])\,0\,(F\,\mathsf{true}_{\mathcal{S}}[\mathsf{p}^2])\,(F\,\mathsf{false}_{\mathcal{S}}[\mathsf{p}^2])$.

Here we check that the P satisfies all the equalities of displayed models. We note that $\sigma_{\mathsf{P}}{}^{\uparrow_{\mathsf{P}}} = \big(\sigma_{\mathsf{P}}[\mathsf{p}\circ F_{\triangleright.1}\circ\mathsf{p},\mathsf{projl}\,\mathsf{q}],\mathsf{projr}\,\mathsf{q}\big)$.

$$
\begin{aligned}
\mathsf{idl}_{\mathsf{P}} &: \mathsf{id}_{\mathsf{P}} \circ_{\mathsf{P}} \sigma_{\mathsf{P}} = 0[F\,\sigma \circ \mathsf{p}, \sigma_{\mathsf{P}}] = \sigma_{\mathsf{P}} \\
\mathsf{idr}_{\mathsf{P}} &: \sigma_{\mathsf{P}} \circ_{\mathsf{P}} \mathsf{id}_{\mathsf{P}} = \sigma_{\mathsf{P}}[F\,\mathsf{id} \circ \mathsf{p}, 0] = \sigma_{\mathsf{P}}[\mathsf{p}, \mathsf{q}] = \sigma_{\mathsf{P}}[\mathsf{id}] = \sigma_{\mathsf{P}} \\
\mathsf{ass}_{\mathsf{P}} &: (\sigma_{\mathsf{P}} \circ_{\mathsf{P}} \delta_{\mathsf{P}}) \circ_{\mathsf{P}} \nu_{\mathsf{P}} = \sigma_{\mathsf{P}}[F\,\delta \circ \mathsf{p}, \delta_{\mathsf{P}}][F\,\nu \circ \mathsf{p}, \nu_{\mathsf{P}}] = \\
&\ \sigma_{\mathsf{P}}[F\,(\delta \circ_{\mathcal{S}} \nu) \circ \mathsf{p}, \delta_{\mathsf{P}}[F\,\nu \circ \mathsf{p}, \nu_{\mathsf{P}}]] = \sigma_{\mathsf{P}} \circ_{\mathsf{P}} (\delta_{\mathsf{P}} \circ_{\mathsf{P}} \nu_{\mathsf{P}}) \\
[\mathsf{id}]_{\mathsf{P}} &: A_{\mathsf{P}}[\mathsf{id}_{\mathsf{P}}]_{\mathsf{P}} = A_{\mathsf{P}}[F\,\mathsf{id} \circ \mathsf{p}^2, \mathsf{q}[\mathsf{p}], \mathsf{q}] = A_{\mathsf{P}}[(\mathsf{p}, \mathsf{q}) \circ \mathsf{p}, \mathsf{q}] = A_{\mathsf{P}}[\mathsf{id} \circ\ \mathsf{p}, \mathsf{q}] = A_{\mathsf{P}}[\mathsf{id}] = A_{\mathsf{P}} \\
[\circ]_{\mathsf{P}} &: A_{\mathsf{P}}[\sigma_{\mathsf{P}} \circ_{\mathsf{P}} \delta_{\mathsf{P}}]_{\mathsf{P}} = A_{\mathsf{P}}[F\,(\sigma \circ_{\mathcal{S}} \delta) \circ \mathsf{p}^2, \sigma_{\mathsf{P}}[F\,\delta \circ \mathsf{p}, \delta_{\mathsf{P}}][\mathsf{p}], \mathsf{q}] = \\
&\ A_{\mathsf{P}}[F\,\sigma \circ \mathsf{p}^2, \sigma_{\mathsf{P}}[\mathsf{p}], \mathsf{q}][F\,\delta \circ \mathsf{p}^2, \delta_{\mathsf{P}}[\mathsf{p}], \mathsf{q}] = A_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}}[\delta_{\mathsf{P}}]_{\mathsf{P}} \\
[\mathsf{id}]_{\mathsf{P}} &: t_{\mathsf{P}}[\mathsf{id}_{\mathsf{P}}]_{\mathsf{P}} = t_{\mathsf{P}}[F\,\mathsf{id} \circ \mathsf{p}, \mathsf{q}] = t_{\mathsf{P}}[\mathsf{p}, \mathsf{q}] = t_{\mathsf{P}}[\mathsf{id}] = t_{\mathsf{P}} \\
[\circ]_{\mathsf{P}} &: t_{\mathsf{P}}[\sigma_{\mathsf{P}} \circ_{\mathsf{P}} \delta_{\mathsf{P}}]_{\mathsf{P}} = t_{\mathsf{P}}[F\,(\sigma \circ \delta) \circ \mathsf{p}, \sigma_{\mathsf{P}}[F\,\delta \circ \mathsf{p}, \delta_{\mathsf{P}}]] = \\
&\ t_{\mathsf{P}}[F\,\sigma \circ \mathsf{p}, \sigma_{\mathsf{P}}][F\,\delta \circ \mathsf{p}, \delta_{\mathsf{P}}] = t_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}}[\delta_{\mathsf{P}}]_{\mathsf{P}} \\
\epsilon\eta_{\mathsf{P}} &: (\delta_{\mathsf{P}} : \mathsf{Sub}_{\mathsf{P}}\,\Gamma_{\mathsf{P}}\,\cdot_{\mathsf{P}}) = (\delta_{\mathsf{P}} : \mathsf{Tm}\,(F\,\Gamma \triangleright \Gamma_{\mathsf{P}})\,\top) \overset{\cdot\eta}{=} \mathsf{tt} = \epsilon_{\mathsf{P}} \\
\triangleright\beta_{1\mathsf{P}} &: \mathsf{p}_{\mathsf{P}} \circ_{\mathsf{P}} (\sigma_{\mathsf{P}} ,_{\mathsf{P}} t_{\mathsf{P}}) = (\mathsf{projl}\,\mathsf{q})[F\,(\sigma ,_{\mathcal{S}} t) \circ \mathsf{p}, (\sigma_{\mathsf{P}}, t_{\mathsf{P}})] = \\
&\ \mathsf{projl}\,(\mathsf{q}[F\,(\sigma ,_{\mathcal{S}} t) \circ \mathsf{p}, (\sigma_{\mathsf{P}}, t_{\mathsf{P}})]) = \mathsf{projl}\,(\sigma_{\mathsf{P}}, t_{\mathsf{P}}) = \sigma_{\mathsf{P}} \\
\triangleright\beta_{2\mathsf{P}} &: \mathsf{q}_{\mathsf{P}}[\sigma_{\mathsf{P}} ,_{\mathsf{P}} t_{\mathsf{P}}]_{\mathsf{P}} = (\mathsf{projr}\,\mathsf{q})[F\,(\sigma ,_{\mathcal{S}} t) \circ \mathsf{p}, (\sigma_{\mathsf{P}}, t_{\mathsf{P}})] = \\
&\ \mathsf{projr}\,(\mathsf{q}[F\,(\sigma ,_{\mathcal{S}} t) \circ \mathsf{p}, (\sigma_{\mathsf{P}}, t_{\mathsf{P}})]) = \mathsf{projr}\,(\sigma_{\mathsf{P}}, t_{\mathsf{P}}) = t_{\mathsf{P}} \\
\triangleright\eta_{\mathsf{P}} &: (\mathsf{p}_{\mathsf{P}} ,_{\mathsf{P}} \mathsf{q}_{\mathsf{P}}) = (\mathsf{projl}\,\mathsf{q}, \mathsf{projr}\,\mathsf{q}) \overset{\Sigma\eta}{=} \mathsf{q} = \mathsf{id}_{\mathsf{P}} \\
,\circ_{\mathsf{P}} &: (\sigma_{\mathsf{P}} ,_{\mathsf{P}} t_{\mathsf{P}}) \circ_{\mathsf{P}} \delta_{\mathsf{P}} = (\sigma_{\mathsf{P}}, t_{\mathsf{P}})[F\,\delta \circ \mathsf{p}, \delta_{\mathsf{P}}] \overset{,[]}{=} (\sigma_{\mathsf{P}}[F\,\delta \circ \mathsf{p}, \delta_{\mathsf{P}}], t_{\mathsf{P}}[F\,\delta \circ \mathsf{p}, \delta_{\mathsf{P}}]) = \\
&\ (\sigma_{\mathsf{P}} \circ_{\mathsf{P}} \delta_{\mathsf{P}} ,_{\mathsf{P}} t_{\mathsf{P}}[\delta_{\mathsf{P}}]_{\mathsf{P}}) \\
\Pi\beta_{\mathsf{P}} &: \mathsf{app}_{\mathsf{P}}\,(\mathsf{lam}_{\mathsf{P}}\,t_{\mathsf{P}}) = \\
&\ \big(\mathsf{app}\,(\mathsf{app}\,(\mathsf{lam}\,(\mathsf{lam}\,(t_{\mathsf{P}}[F_{\triangleright.2} \circ (\mathsf{p}^3, 1), (2, 0)])))))\big) \\
&\ \qquad \big[\mathsf{p} \circ F_{\triangleright.1} \circ \mathsf{p}, \mathsf{projl}\,0, 0[F_{\triangleright.1} \circ \mathsf{p}], \mathsf{projr}\,0\big] \overset{\Pi\beta}{=} \\
&\ t_{\mathsf{P}}[F_{\triangleright.2} \circ (\mathsf{p}^3, 1), (2, 0)][\mathsf{p} \circ F_{\triangleright.1} \circ \mathsf{p}, \mathsf{projl}\,0, 0[F_{\triangleright.1} \circ \mathsf{p}], \mathsf{projr}\,0\big] = \\
&\ t_{\mathsf{P}}[\mathsf{p}, (\mathsf{projl}\,0, \mathsf{projr}\,0)] = t_{\mathsf{P}}[\mathsf{id}] = t_{\mathsf{P}} \\
\Pi\eta_{\mathsf{P}} &: \mathsf{lam}_{\mathsf{P}}\,(\mathsf{app}_{\mathsf{P}}\,t_{\mathsf{P}}) = \\
&\ \mathsf{lam}\,\Big(\mathsf{lam}\,\big(\big(\mathsf{app}\,(\mathsf{app}\,t_{\mathsf{P}})\big)\big[\mathsf{p} \circ F_{\triangleright.1} \circ \mathsf{p}, \mathsf{projl}\,0, 0[F_{\triangleright.1} \circ \mathsf{p}], \mathsf{projr}\,0\big] \\
&\ \qquad\qquad\qquad\qquad \big[F_{\triangleright.2} \circ (\mathsf{p}^3, 1), (2, 0)\big]\big)\Big) = \\
&\ \mathsf{lam}\,\big(\mathsf{lam}\,\big((\mathsf{app}\,(\mathsf{app}\,t_{\mathsf{P}}))[\mathsf{p}^3, 2, 1, 0]\big)\big) = \mathsf{lam}\,(\mathsf{lam}\,((\mathsf{app}\,(\mathsf{app}\,t_{\mathsf{P}}))[\mathsf{id}])) \overset{\Pi\eta_{\mathcal{S}}}{=} t_{\mathsf{P}} \\
\Pi[]_{\mathsf{P}} &: (\Pi_{\mathsf{P}}\,A_{\mathsf{P}}\,B_{\mathsf{P}})[\sigma_{\mathsf{P}}]_{\mathsf{P}} = \\
&\ \Pi\,\big(F\,A[F\,\sigma \circ \mathsf{p}^2]\big) \\
&\ \quad \Big(\Pi\,\big(A_{\mathsf{P}}\big[F\,\sigma \circ \mathsf{p}^2, \sigma_{\mathsf{P}}[\mathsf{p}], \mathsf{q}\big][\mathsf{p}^2, \mathsf{q}]\big) \\
&\ \qquad \big(B_{\mathsf{P}}\big[F_{\triangleright.2} \circ (F\,\sigma \circ \mathsf{p}^4, 1), (\sigma_{\mathsf{P}}[\mathsf{p}^3], 0), \\
&\ \qquad\qquad F\,(\mathsf{app}\,\mathsf{q})[F_{\triangleright.2} \circ (F_{\triangleright.2} \circ (F\,\sigma \circ \mathsf{p}^4, 2), 1)]\big]\big)\Big) = \\
&\ \Pi\,\big(F\,(A[\sigma])[\mathsf{p}^2]\big) \\
&\ \quad \Big(\Pi\,\big(A_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}}[\mathsf{p}^2, \mathsf{q}]\big) \\
&\ \qquad \big(B_{\mathsf{P}}\big[F_{\triangleright.2} \circ (F\,\sigma)^{\uparrow} \circ F_{\triangleright.1} \circ \mathsf{p}^2, \big(\sigma_{\mathsf{P}}[\mathsf{p} \circ F_{\triangleright.1} \circ \mathsf{p}^2, \mathsf{projl}\,1], \mathsf{projr}\,1\big), \mathsf{q}\big] \\
&\ \qquad\qquad \big[F_{\triangleright.2} \circ (\mathsf{p}^4, 1), (3, 0), F\,(\mathsf{app}\,\mathsf{q})[F_{\triangleright.2} \circ (F_{\triangleright.2} \circ (\mathsf{p}^4, 2), 1)]\big]\big)\Big) = \\
&\ \Pi_{\mathsf{P}}\,(A_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}})\,(B_{\mathsf{P}}[{\sigma_{\mathsf{P}}}^{\uparrow_{\mathsf{P}}}]_{\mathsf{P}})
\end{aligned}
$$

$$
\begin{aligned}
\mathsf{lam}[]_{\mathsf{P}} \quad &:(\mathsf{lam}_{\mathsf{P}}\, t_{\mathsf{P}})[\sigma_{\mathsf{P}}]_{\mathsf{P}} = \mathsf{lam}\,(\mathsf{lam}\,(t_{\mathsf{P}}[F_{\triangleright.2} \circ (F\,\sigma \circ \mathsf{p}^3, 1), (\sigma_{\mathsf{P}}[\mathsf{p}^2], 0)])) = \\
&\quad \mathsf{lam}_{\mathsf{P}}\,(t_{\mathsf{P}}[\sigma_{\mathsf{P}}{}^{\uparrow_{\mathsf{P}}}]_{\mathsf{P}}) \\
\Sigma\beta_{1\mathsf{P}} \quad &:\mathsf{projl}_{\mathsf{P}}\,(u_{\mathsf{P}} ,_{\mathsf{P}} v_{\mathsf{P}}) = \mathsf{projl}\,(u_{\mathsf{P}}, v_{\mathsf{P}}) = u_{\mathsf{P}} \\
\Sigma\beta_{2\mathsf{P}} \quad &:\mathsf{projr}_{\mathsf{P}}\,(u_{\mathsf{P}} ,_{\mathsf{P}} v_{\mathsf{P}}) = \mathsf{projr}\,(u_{\mathsf{P}}, v_{\mathsf{P}}) = v_{\mathsf{P}} \\
\Sigma\eta_{\mathsf{P}} \quad &:(\mathsf{projl}_{\mathsf{P}}\, t_{\mathsf{P}} ,_{\mathsf{P}} \mathsf{projr}_{\mathsf{P}}\, t_{\mathsf{P}}) = (\mathsf{projl}\, t_{\mathsf{P}}, \mathsf{projr}\, t_{\mathsf{P}}) = t_{\mathsf{P}} \\
\Sigma[]_{\mathsf{P}} \quad &:(\Sigma_{\mathsf{P}}\, A_{\mathsf{P}}\, B_{\mathsf{P}})[\sigma_{\mathsf{P}}]_{\mathsf{P}} = \\
&\quad \Sigma\,\big(A_{\mathsf{P}}\big[F\,\sigma \circ \mathsf{p}^2, \sigma_{\mathsf{P}}[\mathsf{p}], F\,(\mathsf{projl}\,\mathsf{q})[F_{\triangleright.2} \circ (F\,\sigma \circ \mathsf{p}^2, \mathsf{q})]\big]\big) \\
&\qquad \big(B_{\mathsf{P}}\big[F_{\triangleright.2} \circ \big(F\,\sigma \circ \mathsf{p}^3, F\,(\mathsf{projl}\,\mathsf{q})[F_{\triangleright.2} \circ (F\,\sigma \circ \mathsf{p}^3, 1)]\big), (\sigma_{\mathsf{P}}[\mathsf{p}^2], 0), \\
&\qquad\qquad F\,(\mathsf{projr}\,\mathsf{q})[F_{\triangleright.2} \circ (F\,\sigma \circ \mathsf{p}^3, 1)]\big]\big) = \\
&\quad \Sigma\,\big(A_{\mathsf{P}}\big[F\,\sigma \circ \mathsf{p}^2, \sigma_{\mathsf{P}}[\mathsf{p}], F\,(\mathsf{projl}\,\mathsf{q})[F_{\triangleright.2} \circ (\mathsf{p}^2, \mathsf{q})]\big]\big) \\
&\qquad \big(B_{\mathsf{P}}\big[F_{\triangleright.2} \circ \big(F\,\sigma \circ \mathsf{p}^3, F\,(\mathsf{projl}\,\mathsf{q})[F_{\triangleright.2} \circ (\mathsf{p}^3, 1)]\big), (\sigma_{\mathsf{P}}[\mathsf{p}^2], 0), \\
&\qquad\qquad F\,(\mathsf{projr}\,\mathsf{q})[F_{\triangleright.2} \circ (\mathsf{p}^3, 1)]\big]\big) = \\
&\quad \Sigma_{\mathsf{P}}\,(A_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}})\,(B_{\mathsf{P}}[\sigma_{\mathsf{P}}{}^{\uparrow_{\mathsf{P}}}]_{\mathsf{P}}) \\
,[]_{\mathsf{P}} \quad &:(u_{\mathsf{P}} ,_{\mathsf{P}} v_{\mathsf{P}})[\sigma_{\mathsf{P}}]_{\mathsf{P}} = (u_{\mathsf{P}}[F\,\sigma \circ \mathsf{p}, \sigma_{\mathsf{P}}], v_{\mathsf{P}}[F\,\sigma \circ \mathsf{p}, \sigma_{\mathsf{P}}]) = (u_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}} ,_{\mathsf{P}} v_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}}) \\
\top\eta_{\mathsf{P}} \quad &:(t_{\mathsf{P}} : \mathsf{Tm}_{\mathsf{P}}\,\Gamma_{\mathsf{P}}\,\top_{\mathsf{P}}) = (t_{\mathsf{P}} : \mathsf{Tm}\,(F\,\Gamma \triangleright \Gamma_{\mathsf{P}})\,\top) \overset{\top\eta}{=} \mathsf{tt} = \mathsf{tt}_{\mathsf{P}} \\
\top[]_{\mathsf{P}} \quad &:\top_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}} = \top = \top_{\mathsf{P}} \\
\mathsf{tt}[]_{\mathsf{P}} \quad &:\mathsf{tt}_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}} = \mathsf{tt} = \mathsf{tt}_{\mathsf{P}} \\
\mathsf{U}\beta_{\mathsf{P}} \quad &:\mathsf{El}_{\mathsf{P}}\,(\mathsf{c}_{\mathsf{P}}\, A_{\mathsf{P}}) = \mathsf{El}\,(\mathsf{app}\,(\mathsf{lam}\,(\mathsf{c}\, A_{\mathsf{P}}))) \overset{\Pi\beta}{=} \mathsf{El}\,(\mathsf{c}\, A_{\mathsf{P}}) \overset{\mathsf{U}\beta}{=} A_{\mathsf{P}} \\
\mathsf{U}\eta_{\mathsf{P}} \quad &:\mathsf{c}_{\mathsf{P}}\,(\mathsf{El}_{\mathsf{P}}\, a_{\mathsf{P}}) = \mathsf{lam}\,(\mathsf{c}\,(\mathsf{El}\,(\mathsf{app}\, a_{\mathsf{P}}))) \overset{\mathsf{El}\eta}{=} \mathsf{lam}\,(\mathsf{app}\, a_{\mathsf{P}}) \overset{\Pi\eta}{=} a_{\mathsf{P}} \\
\mathsf{U}[]_{\mathsf{P}} \quad &:(\mathsf{U}_{\mathsf{P}}\, i)[\sigma_{\mathsf{P}}]_{\mathsf{P}} = F\,(\mathsf{El}_{\mathcal{S}}\,\mathsf{q})[F_{\triangleright.2} \circ (F\,\sigma \circ \mathsf{p}^2, \mathsf{q})] \Rightarrow \mathsf{U}\, i = \\
&\quad F\,(\mathsf{El}_{\mathcal{S}}\,\mathsf{q})[F_{\triangleright.2} \circ (F\,\sigma)^{\uparrow} \circ (\mathsf{p}^2, \mathsf{q})] \Rightarrow \mathsf{U}\, i = \\
&\quad F\,(\mathsf{El}_{\mathcal{S}}\,\mathsf{q})[F\,(\sigma^{\uparrow}) \circ F_{\triangleright.2} \circ (\mathsf{p}^2, \mathsf{q})] \Rightarrow \mathsf{U}\, i = F\,(\mathsf{El}_{\mathcal{S}}\,\mathsf{q})[F_{\triangleright.2} \circ (\mathsf{p}^2, \mathsf{q})] \Rightarrow \mathsf{U}\, i = \mathsf{U}_{\mathsf{P}}\, i \\
\mathsf{El}[]_{\mathsf{P}} \quad &:(\mathsf{El}_{\mathsf{P}}\, a_{\mathsf{P}})[\sigma_{\mathsf{P}}]_{\mathsf{P}} = (\mathsf{El}\,(\mathsf{app}\, a_{\mathsf{P}}))[F\,\sigma \circ \mathsf{p}^2, \sigma_{\mathsf{P}}[\mathsf{p}], \mathsf{q}] \overset{\mathsf{El}[]}{=} \\
&\quad \mathsf{El}\,((\mathsf{app}\, a_{\mathsf{P}})[F\,\sigma \circ \mathsf{p}^2, \sigma[\mathsf{p}], \mathsf{q}]) \overset{\mathsf{app}[]}{=} \mathsf{El}\,(\mathsf{app}\,(a_{\mathsf{P}}[F\,\sigma \circ \mathsf{p}, \sigma_{\mathsf{P}}])) = \mathsf{El}_{\mathsf{P}}\,(a_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}}) \\
\mathsf{Bool}[]_{\mathsf{P}} \quad &:\mathsf{Bool}_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}} = \\
&\quad \Sigma\,\mathsf{Bool}\,\big(\mathsf{Id}\,(F\,\mathsf{Bool}[F\,\sigma \circ \mathsf{p}^3]) \\
&\qquad\qquad \big(\mathsf{if}\,(F\,\mathsf{Bool}[F\,\sigma \circ \mathsf{p}^4])\, 0\, (F\,\mathsf{true}[F\,\sigma \circ \mathsf{p}^3])\, (F\,\mathsf{false}[F\,\sigma \circ \mathsf{p}^3])\big)\, 1\big) = \\
&\quad \Sigma\,\mathsf{Bool}\,\big(\mathsf{Id}\,(F\,\mathsf{Bool}[\mathsf{p}^3])\,\big(\mathsf{if}\,(F\,\mathsf{Bool}[\mathsf{p}^4])\, 0\, (F\,\mathsf{true}[\mathsf{p}^3])\, (F\,\mathsf{false}[\mathsf{p}^3])\big)\, 1\big) = \mathsf{Bool}_{\mathsf{P}} \\
\mathsf{true}[]_{\mathsf{P}} \quad &:\mathsf{true}_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}} = \big(\mathsf{true}, \mathsf{refl}\,(F\,(\mathsf{true}_{\mathcal{S}}[\sigma])[\mathsf{p}])\big) \overset{\mathsf{true}[]_{\mathcal{S}}}{=} (\mathsf{true}, \mathsf{refl}\,(F\,\mathsf{true}_{\mathcal{S}}[\mathsf{p}])) = \mathsf{true}_{\mathsf{P}} \\
\mathsf{false}[]_{\mathsf{P}} \quad &:\mathsf{false}_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}} = \big(\mathsf{false}, \mathsf{refl}\,(F\,(\mathsf{false}_{\mathcal{S}}[\sigma])[\mathsf{p}])\big) \overset{\mathsf{false}[]_{\mathcal{S}}}{=} (\mathsf{false}, \mathsf{refl}\,(F\,\mathsf{false}_{\mathcal{S}}[\mathsf{p}])) = \mathsf{false}_{\mathsf{P}} \\
\mathsf{if}[]_{\mathsf{P}} \quad &:(\mathsf{if}_{\mathsf{P}}\,_\, t_{\mathsf{P}}\, u_{\mathsf{P}}\, v_{\mathsf{P}})[\sigma_{\mathsf{P}}]_{\mathsf{P}} = \\
&\quad \big(\mathsf{J}\,_\,(\mathsf{if}\,_\,(\mathsf{projl}\, t_{\mathsf{P}})\, u_{\mathsf{P}}\, v_{\mathsf{P}})\,(\mathsf{projr}\, t_{\mathsf{P}})\big)[F\,\sigma \circ \mathsf{p}, \sigma_{\mathsf{P}}] \overset{\mathsf{J}[],\mathsf{if}[],\mathsf{projl}[],\mathsf{projr}[]}{=} \\
&\quad \big(\mathsf{J}\,_\,(\mathsf{if}\,_\,(\mathsf{projl}\,(t_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}}))\,(u_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}})\,(v_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}}))\,(\mathsf{projr}\,(t_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}}))\big) = \\
&\quad \mathsf{if}_{\mathsf{P}}\,_\,(t_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}})\,(u_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}})\,(v_{\mathsf{P}}[\sigma_{\mathsf{P}}]_{\mathsf{P}})
\end{aligned}
$$