Stage de M1

École Normale Supérieure
Département d'Informatique

Modèles de la théorie des types dépendants et de l'axiome d'univalence

Rapport de stage

David REBOULLET
david.reboullet@ens.fr

Février-Juillet 2018

Sous la direction de
Thierry Coquand
coquand@chalmers.se

Computer Science and Engineering Department
University of Gothenburg
SE-412 96 Göteborg - Sweden

Contents

Introduction **1**

1 Semantic of an extensional type theory in presheaves **1**
1.1 Syntax . . . 3
1.2 Interpretation of the syntax . . . 6
1.3 Derivation rules . . . 11
1.4 Discussion . . . 12

2 Extension structure and the boolean stack model **13**
2.1 Extension structure in dependent type theory . . . 13
2.2 Extension structure in cubical type theory . . . 15
2.3 The boolean stack model . . . 16

Conclusion **19**

References **19**

A Proofs from section 1 **21**
A.1 Semantic weakening . . . 21
A.2 Semantic substitution . . . 23
A.3 Derivation rules . . . 24

B Proofs from section 2 **29**
B.1 Extension structure in dependent type theory . . . 29
B.2 Extension structure in cubical type theory . . . 30

Introduction

The five months internship of the second year of computer science at the ENS was for me an opportunity to get a taste of the research in logic and computer science. To that extent I contacted Thierry Coquand at the computer science and engineering department of the University of Gothenburg and got a chance to face modern issues in the field of homotopy type theory and univalent foundations.

Univalent foundations [Voe10] are based on a new axiom, the Univalence Axiom, formulated by Vladimir Voevodsky inside Martin-Löf Type Theory, or MLTT, [ML79] in order to close the gap between classical pure mathematics and computer verification. Nevertheless this axioms at first lacks computational contents, which prevents in general one to get a value out of a proof using it. This issue has led to the creation of the cubical type theory, or CTT, [CCHM16] which is a constructive extension of MLTT in which the univalence axiom is provable. The proof of coherence of this theory comes from the study of the presheaves on a category called the cube category, but it was shown in [OP17] that we can generalize the construction of suitable models of CTT by working internally in a category satisfying some axioms. Thanks to this generalization one could study models of CTT that satisfy additional properties. For instance, we will be interested in this report by the compatibility of CTT with intuitionistic principles. We see this idea of using an internal language to make models of CTT as a huge improvement from other approaches as for instance the proof of the fibrancy of Π-types fits in a single page in [OP17] compared to the 20 pages and highly technical proof found in [GS17]. Nevertheless [OP17] lacks a formal definition and justification of the language they use, and this could cast legitimate doubts on the validity of their developments. As far as we know, the formal construction of internal dependent type theories as only been done for intensional languages inside toposes [Str91] and for extensional languages inside set theories [Acz98, Bar10], extensional being more general than intensional and toposes being more general than sets. The issue with [OP17] is then that it uses an extensional language to study toposes, but the semantics of such theory has not been studied yet.

The objective of this internship is to prove the mutual coherence of the univalence axiom and Brouwer's fan theorem. The latter despite being true in a classical setting does not always hold in constructive mathematics. This result would allow one to do point-wise topology in a univalent type theory. To that extend we reduce thanks to [OP17] the semantic of our theory where both univalence and the fan theorem hold to the semantic of an extensional type theory with added axioms. We give such semantics in the first part of this report. Then in the second part we build a model of these two axioms.

1 Semantic of an extensional type theory in presheaves

The issue that prevents us from relying on [Str91] to provide a justification of the internal language used in [OP17] is that it crucially relies on the strong normalization of the calculus under scrutiny. Indeed to obtain a well defined interpretation of terms Streicher adds annotations to his calculus and then prove thanks to strong normalization that such annotations always exist for well-typed terms and are unique. As we want to formalize the use of an extensional type theory we cannot assume strong normalization. As an extensional type theory is a type theory where the equality type implies the definitional equality, that is to say that a variable of equality type in scope adds the corresponding equivalence during type checking, it is easy to see that such theories cannot have strong normalization. For instance the following term reduces to itself :

$$\lambda\,(\_ : \mathbb{N} \equiv_{\mathcal{U}} (\mathbb{N} \to \mathbb{N}))\,(\lambda\,(x : \mathbb{N})\ x\ x)\,(\lambda\,(x : \mathbb{N})\ x\ x)$$

On the other hand Aczel manages to give in [Acz98] a semantic of an extensional type theory in sets. His methodology is to first define the interpretation of a term with no respects to its typability or derivation rules. Then he translates statements of the theory into (meta-theoretic) statements about the interpretation of terms. Proving the rules of the type theory is then to prove some semantic statements one by one in isolation. The big advantage of such semantics is that we never need to deal with the whole picture. Nevertheless Aczel relies on usual encodings to get his semantic and extending it to presheaf models, the usual models of HOTT and CTT, seems too technical. Instead, we keep the idea of building the full semantic step-by-step but we would like to gather the conditions under which the interpretation of a term is defined *while* we define it in order to keep the definitions manageable. The issue here is that the conditions for one term may depend on the interpretation of other terms, hence the definition of the conditions and the interpretations are mutual. Such simultaneous definitions are not always well defined themselves, but in our case it falls into the pattern of simultaneous inductive-recursive definitions proven correct in [Dyb00].

A category $\mathcal{C}$ is a collection $Ob\,(\mathcal{C})$, which is usually just denoted by $\mathcal{C}$, and for each $x, y \in \mathcal{C}$ a collection of “ morphisms ” $\mathcal{C}\,(x, y)$ such that for $f \in \mathcal{C}\,(x, y)$ and $g \in \mathcal{C}\,(y, z)$ we have a composition operation $g \circ f \in \mathcal{C}\,(x, z)$ which is associative and which has a left and right neutral id_x for each $x \in \mathcal{C}$. A presheaf P on $\mathcal{C}$ is the data of a set $P\,(x)$ for each $x \in \mathcal{C}$ and a function $P\,(f) : P\,(y) \to P\,(x)$ for each element $f \in \mathcal{C}\,(y, x)$ such that the composition in $\mathcal{C}$ in sent by P to the composition of set theoretic functions, and that preserves neutrals. Note the reversal of the arrows. We often call an element of the image of a function $P\,(f)$ a restriction along f and denote it $\rho \cdot f$ in $\rho \in dom\,(P\,(f))$. A morphism $N : P \to Q$ of presheaves is for each $x \in \mathcal{C}$ a function $N\,(x) : P\,(x) \to P\,(y)$ such that for each morphism $f : y \to x$ in $\mathcal{C}$ we have $N\,(y) \circ P\,(f) = Q\,(f) \circ N\,(x)$. This makes the presheaves over $\mathcal{C}$ a category which we denote by $\widehat{\mathcal{C}}$. Presheaves can be thought as a generalization of indexed sets. Moreover, by taking $\mathcal{C}$ to be a category to be a singleton with no non neutral morphisms we see that sets are a special case of presheaves. In fact they also provide models of at least intuitionistic logic, see [MM94].

A type theory is a formal system for deriving statements of the form $\Gamma \vdash \mathcal{J}$ where Γ is a context that maps variable names to their type (in the usual computer scientific sens) and $\mathcal{J}$ is the statement body. For instance $\mathcal{J}$ may state that a term has some type or that two terms are equal. Equality is there to be understood as an undirected reducibility relation.

Until the end of this section we will give ourself a small category $\mathcal{C}$. Our aim will be to interpret an extensional MLTT with a universe of strict propositions and propositional resizing inside $\widehat{\mathcal{C}}$. As a side condition, the interpretation of a type under the empty context should be an object of $\widehat{\mathcal{C}}$ with the usual categorical properties we would expect from the type. For instance the interpretation of a Π-type should be an exponential. A term of a type should be a global element of the corresponding presheaf.

The set theoretic encoding of numbers as ordinals will be used without warnings. We will assume that our set meta-theory has a hierarchy of $\omega+1$ Grothendieck universes that we will denote V_n, $n \in \omega + 1$. Each Grothendieck universe is a set which is by definition big enough to be stable by each set theoretic operation, hence we can only postulate them. We will only consider presheaves with codomain V_ω.

We will denote the projections using the letter π. We will take the unusual convention of having the projections numbered from 0 and starting from the last element of a tuple. This will avoid unnecessary arithmetics when dealing with De Bruijn indexes. If P is a proposition on indexes will denote by π_P the projection along the indexes satisfying P. We will also denote by $\hat{\pi}_X$ the complementary projection to π_X whether X is an index or a proposition. Also, the (unusual and personal) notation $f \setminus g$ will be used to name the arrow from $f \circ g$ to f entailed by g in the slice category of $\mathcal{C}$ which is over the codomain of f.

The work in this section is personal, if we omit all the discussions and suggestions I had from my supervisor.

1.1 Syntax

We will start by defining the syntax of the type theory we consider. We will denote variables by DeBruijn indexes to avoid issues related with renaming, hence a term var_i will denote the i-th inner variable binding. Thanks to this notation we no longer need to specify the name of a variable in a quantifier and their first parameter will always be the type abstracted over. For instance the lambda abstraction $\lambda\,(x : A)\,B$ becomes $\lambda\ A\ B$. Moreover as DeBruijn terms represent equivalent classes of " regular " terms modulo renaming this change of syntax is valid to interpret the usual notation.

$$
\begin{array}{rcll}
term_n & := & var_i & i < n \\
& \mid & \Pi\ term_n\ term_{n+1} & \\
& \mid & \lambda\ term_n\ term_{n+1} & \\
& \mid & term_n\ term_n & \\
& \mid & \Sigma\ term_n\ term_{n+1} & \\
& \mid & term_n, term_n & \\
& \mid & term_n.1 & \\
& \mid & term_n.2 & \\
& \mid & \mathcal{U}_l & l \in \mathbb{N} \\
& \mid & \Omega & \\
& \mid & * &
\end{array}
$$

$$\begin{array}{ll} | & term_n \equiv_{term_n} term_n \\ | & \|term_n\| \\ | & |term_n| \\ | & \langle term_n \rangle \end{array}$$

As usual we will denote function types by Π, function abstraction by λ and application with the functional notation. Pair types will be denoted by Σ, with M, N as constructor and $P.1$ and $P.2$ being the projections. $\mathcal{U}_l, l \in \mathbb{N}$ will be a hierarchy of impredicative universes (ie. types of types). Ω will the impredicative universe of propositions, with $*$ being the unique inhabitant of a true proposition. $M \equiv_A N$ is the proposition stating that two terms M and N of type A are equal. For A a type $\|A\|$ is the proposition stating that A is inhabited (ie. the propositional truncation of A). We also call such types *squashed types*. We denote by $|M|$ the introduction of a squashed type. $\langle N \rangle$ is the eliminator of squashed types. It states that constant functions factor through the propositional truncation of their domain.

We then define the $context_n$, $n \in \mathbb{N}$, of well-formed contexts of length n. We will usually name contexts Γ. We also define $context := \bigcup_{n \in \mathbb{N}} context_n$ and $|\Gamma \in context| := n$ when $\Gamma \in context_n$.

$$\begin{array}{lll} context_0 & := & \epsilon \\ context_{n+1} & := & context_n,\ term_n \end{array}$$

We define a telescope over a context of length $n \in \mathbb{N}$, and denote their set by $telescope_n$, as being a context whose base case is any context of length n. Telescopes will be named with the letter Δ. Given a telescope $\Delta \in telescope_n$ we denote by $|\Delta|$ the length of Δ, defined as the length of contexts. This naturally define a composition operation that given a context $\Gamma \in context_n$, $n \in \mathbb{N}$, and a telescope $\Delta \in telescope_n$ gives a context $\Gamma, \Delta \in context_{n+|\Delta|}$. This formalizes the usual pratice of splitting contexts.

$$\begin{array}{lll} telescope_n & := & \bigsqcup_{k \in \mathbb{N}} telescope_n^k \\ telescope_n^0 & := & \epsilon_n \\ telescope_n^{k+1} & | & telescope_n^k,\ term_{n+k} \end{array}$$

We define the syntax of judgments under a context of length n.

$$\begin{array}{lll} judgment_n & := & \text{ok} \\ & | & term_n\ type \\ & | & term_n : term_n \\ & | & term_n = term_n\ type \\ & | & term_n = term_n : term_n \end{array}$$

A sequent is then a context and a judgment separated by $\vdash$. The judgement ok means that the context is valid, the colon denotes the membership of a type and the equal sign the definitional equality. We will often forget to write the judgment in the sequent $\Gamma \vdash$ ok, that will be written $\Gamma \vdash$.

$$sequent \ := \ context_n \ \vdash \ judgment_n,\ n \in \mathbb{N}$$

Weakening We define for all $n, k \in \mathbb{N}$, $k \leq n$, a weakening operator $\uparrow_k$: $term_n \to term_{n+1}$ by induction on the term.

$$
\begin{array}{lcl}
\uparrow_k \ var_i,\ i < k & := & var_i \\
\uparrow_k \ var_i,\ i \geq k & := & var_{i+1} \\
\uparrow_k \ \Pi\ A\ M & := & \Pi\ (\uparrow_k A)\ (\uparrow_{k+1} M) \\
\uparrow_k \ \lambda\ A\ M & := & \lambda\ (\uparrow_k A)\ (\uparrow_{k+1} M) \\
\uparrow_k \ M\ N & := & (\uparrow_k M)\ (\uparrow_k N) \\
\uparrow_k \ \Sigma\ A\ M & := & \Sigma\ (\uparrow_k A)\ (\uparrow_{k+1} M) \\
\uparrow_k \ M, N & := & (\uparrow_k M), (\uparrow_k N) \\
\uparrow_k \ M.1 & := & (\uparrow_k M).1 \\
\uparrow_k \ M.2 & := & (\uparrow_k M).2 \\
\uparrow_k \ \mathcal{U}_l & := & \mathcal{U}_l \\
\uparrow_k \ \Omega & := & \Omega \\
\uparrow_k \ * & := & * \\
\uparrow_k \ M \equiv_A N & := & (\uparrow_k M) \equiv_{\uparrow_k A} (\uparrow_k N) \\
\uparrow_k \ \|M\| & := & \|\uparrow_k M\| \\
\uparrow_k \ |M| & := & |\uparrow_k M| \\
\uparrow_k \ \langle M \rangle & := & \langle \uparrow_k M \rangle
\end{array}
$$

We then define the weakening $\uparrow \Delta \in telescope_{n+1}$ of a telescope $\Delta \in telescope_n$, where $n \in \mathbb{N}$.

$$
\begin{array}{lcl}
\uparrow \epsilon_n & := & \epsilon_{n+1} \\
\uparrow (\Delta, M) & := & \uparrow \Delta,\ \uparrow_{|\Delta|} M
\end{array}
$$

We also define the weakening $\uparrow_k \mathcal{J} \in judgement_{n+1+k}$ for $\mathcal{J} \in judgement_{n+k}$ the obvious way, with $n, k \in \mathbb{N}$.

Substitution With this we can define by induction a substitution operator $-[k \setminus -] : term_{n+1+k} \to term_n \to term_{n+k}$ for all $n, k \in \mathbb{N}$.

$$
\begin{array}{lrcl}
var_0 & [0 \setminus S] & := & S \\
var_{i+1} & [0 \setminus S] & := & var_i \\
var_0 & [k+1 \setminus S] & := & var_0 \\
var_{i+1} & [k+1 \setminus S] & := & \uparrow_0 (var_i\,[k \setminus S]) \\
\Pi\ A\ M & [k \setminus S] & := & \Pi\ (A\,[k \setminus S])\ (M\,[k+1 \setminus S]) \\
\lambda\ A\ M & [k \setminus S] & := & \lambda\ (A\,[k \setminus S])\ (M\,[k+1 \setminus S]) \\
M\ N & [k \setminus S] & := & (M\,[k \setminus S])\ (N\,[k \setminus S]) \\
\Sigma\ A\ M & [k \setminus S] & := & \Sigma\ (A\,[k \setminus S])\ (M\,[k+1 \setminus S]) \\
M, N & [k \setminus S] & := & (M\,[k \setminus S]), (N\,[k \setminus S]) \\
M.1 & [k \setminus S] & := & (M\,[k \setminus S]).1 \\
M.2 & [k \setminus S] & := & (M\,[k \setminus S]).2 \\
\mathcal{U}_l & [k \setminus S] & := & \mathcal{U}_l \\
\Omega & [k \setminus S] & := & \Omega
\end{array}
$$

$$\begin{array}{lll} * \quad [k \setminus S] & := & * \\ M \equiv_A N \quad [k \setminus S] & := & (M\,[k \setminus S]) \equiv_{A[k\setminus A]} (N\,[k \setminus S]) \\ \|M\| \quad [k \setminus S] & := & \|M\,[k \setminus S]\| \\ |M| \quad [k \setminus S] & := & |M\,[k \setminus S]| \\ \langle M \rangle \quad [k \setminus S] & := & \langle M\,[k \setminus S] \rangle \end{array}$$

Like the weakening we define a substitution $\Delta\,[S] \in telescope_n$ under a telescope $\Delta \in telescope_{n+1}$ by a term $S \in term_n$.

$$\begin{array}{lll} \epsilon_{n+1}\,[S] & := & \epsilon_n \\ \Delta, M\,[S] & := & \Delta\,[S]\,, M\,[|\Delta| \setminus S] \end{array}$$

Finally we define the substitution $\mathcal{J}\,[k \setminus S] \in judgment_{n+k}$ of a judgment $\mathcal{J} \in judgment_{n+1+k}$ by a term $S \in term_n$ the obvious way, with $n, k \in \mathbb{N}$.

1.2 Interpretation of the syntax

Semantic definitions We first define the semantic counterparts of the syntactic definitions above. A semantic context of length $n \in \mathbb{N}$ is a sub-object of a n-product in $\widehat{\mathcal{C}}$ and denote the set of such objects by $\widehat{\mathcal{C}_n}$. That is to say that whenever $\xi \in \widehat{\mathcal{C}_n}$ then for each $x \in \mathcal{C}$ elements of ξ_x are n-tuples. A semantic term under a context $\xi \in \widehat{\mathcal{C}_n}$ should be a morphism from ξ to some object of $\widehat{\mathcal{C}}$. But we don't want to specify the codomain of a term in its definition as we want to allow a term to be a member of different types. Thus a semantic term becomes a mapping from $(x \in \mathcal{C}) \times \xi_x$ to V_ω, and we denote their set by $term_\xi$.

We then want to isolate a subset $type_\xi$ of $term_\xi$ witch correspond to the terms that can be interpreted as types under ξ. In the terms of category theory this would mean that a term $\alpha \in type_\xi$ is an element of some object classifier, depending of its set theoretic size. Nevertheless we want to free the definition of a semantic type of any size issue so we take the explicit lifting of Grothendieck universes to object classifiers of $\widehat{\mathcal{C}}$ found in [HS97] and apply it to V_ω. The intuition is that we should get a presheaf which, despite being too big to be a member of $\widehat{\mathcal{C}}$, generalizes the membership of any object classifier in $\widehat{\mathcal{C}}$. Concretely we get the following definition of $type_\xi$, where $f^* : \widehat{\mathcal{C}/x} \to \widehat{\mathcal{C}/y}$ is induced by $f : y \to x$ by selecting the morphisms that factor through f :

$$\begin{array}{lll} \alpha \in type_\xi & \iff & \alpha \in term_\xi \\ & \& & \forall x \in \mathcal{C}, \forall \rho \in \xi_x,\ \alpha\,(x, \rho) \in \widehat{\mathcal{C}/x} \\ & \& & \forall x, y \in \mathcal{C}, \forall f \in \mathcal{C}\,(y, x)\,, \forall \rho \in \xi_x,\ \alpha\,(y, \rho \cdot f) = f^*\,(\alpha\,(x, \rho)) \end{array}$$

Given $\alpha \in type_\xi$ we can now define the set $term_{\xi;\alpha} \subset term_\xi$ of elements of α.

$$\begin{array}{lll} \beta \in term_{\xi;\alpha} & \iff & \beta \in term_\xi \\ & \& & \forall x \in \mathcal{C}, \forall \rho \in \xi_x,\ \beta\,(x, \rho) \in \alpha\,(x, \rho)\,(id_x) \\ & \& & \forall x, y \in \mathcal{C}, \forall f \in \mathcal{C}\,(y, x)\,, \forall \rho \in \xi_x, \\ & & \beta\,(y, \rho \cdot f) = \alpha\,(x, \rho)\,(id_x \backslash f)\,(\beta\,(x, \rho)) \end{array}$$

We can also define the context extension $\xi, \alpha \in \widehat{\mathcal{C}}_{n+1}$ of $\xi \in \widehat{\mathcal{C}}_n$ by $\alpha \in type_\xi$, where $n \in \mathbb{N}$. This corresponds to the syntactic context weakening rule and likewise it allows us to construct contexts from semantic types.

$$\begin{array}{lll} \xi, \alpha\,(x \in \mathcal{C}) & := & \{\rho, a \mid \rho \in \xi_x,\ a \in \alpha\,(x, \rho)\} \\ \xi, \alpha\,(f \in \mathcal{C}\,(y, x)) & := & (\rho, a) \mapsto (\rho \cdot f,\ \alpha\,(x, \rho)\,(id_x \backslash f)\,(a)) \end{array}$$

It should be noted that if we set $\xi \in \widehat{\mathcal{C}}_n$ then the previous construction provide an isomorphism between $type_\xi$ and contexts $\xi' \in \widehat{\mathcal{C}}_{n+1}$ such that $\hat{\pi}_0\,\xi' = \xi$. For each $x \in \mathcal{C}$, $\rho \in \xi_x$, taking the fiber of ξ' over ξ induced by ρ considered as a partial element and then reindexing the resulting presheaf by the forgetful functor from $(C/x)^{op}$ to $\mathcal{C}^{op}$ gives the inverse, but a less abstract definition of this construction is also easily seen. Because we can present any presheaf F over ξ as a context extension of ξ this makes the link between our definition and the usual category theoretic definition of types and contexts.

Term's interpretation We define for all $n \in \mathbb{N}$, $\xi \in \widehat{\mathcal{C}}_n$, an interpretation function $[\![-]\!]_\xi : term_n \to term_\xi$ for each term which satisfy the predicate $[-]_\xi$. Here $[\![-]\!]$ is defined by recursion on terms and $[-]$ is defined as an indexed inductive type. They are mutualy defined, the correctness of the construction comes from [Dyb00]. We will avoid explicit quantification in the following definitions, for instance each of them will quantify over objects $x, y, z \in \mathcal{C}$, morphisms $f \in \mathcal{C}\,(y, x)$, $g \in \mathcal{C}\,(z, y)$ and elements $\rho \in \xi_x$.

For variables we simply take the projection corresponding to the DeBruijn index. This is always defined by definition of a semantic context.

$$\begin{array}{lll} [var_i]_\xi & := & \top \\ [\![var_i]\!]_\xi\,(x, \rho) & := & \pi_i\,(\rho) \end{array}$$

For Π-types we ask the domain and codomain interpretation to give semantic types. We would like the semantic of $\Pi\ A\ B$ as stage x with $\rho \in \xi_x$ to be

$$\left(a \in [\![A]\!]_\xi\,(x, \rho)\,(id_x)\right) \mapsto [\![B]\!]_{\xi, [\![A]\!]_\xi}\,(x, (\rho, a))\,(id_x)$$

but as for regular exponential it would be impossible to define suitable morphisms between stages for this type. Still like for regular exponentials the fix is to add a quantification over objects under x and to define morphisms between stages. Moreover, this define the Π-type at one stage but to get a semantic type at this stage we also need to provide definitions for each stages under the current one. This makes the following definition :

$$\begin{array}{lll} [\Pi\ A\ B]_\xi & := & [A]_\xi\ \&\ [\![A]\!]_\xi \in type_\xi\ \&\ [B]_{\xi, [\![A]\!]_\xi}\ \&\ [\![B]\!]_{\xi, [\![A]\!]_\xi} \in type_{\xi, [\![A]\!]_\xi} \\ [\![\Pi\ A\ B]\!]_\xi\,(x, \rho)\,(f) & := & \\ \multicolumn{3}{l}{\{\alpha \in (g \in \Sigma z \mathcal{C}\,(z, y)) \to (a \in [\![A]\!]_\xi\,(z, \rho \cdot f \cdot g)\,(id_z)) \to [\![B]\!]_{\xi, [\![A]\!]_\xi}\,(z, (\rho \cdot f \cdot g, a))\,(id_z)} \\ \multicolumn{3}{r}{\mid \forall g \in \Sigma z \mathcal{C}\,(z, y)\,,\ h \in \Sigma w \mathcal{C}\,(w, z)\,,\ \forall a,\ \alpha\,(f)\,(a) \cdot g = \alpha\,(g \circ f)\,(a \cdot g)\}} \\ [\![\Pi\ A\ B]\!]_\xi\,(x, \rho)\,(f \backslash g) & := & e \mapsto (h \in \Sigma w \mathcal{C}\,(w, z)) \mapsto e\,(g \circ h) \end{array}$$

From there λ-abstraction and application are immediate. Hopefully we lose a level of indirection at each step.

$$\begin{array}{lll}
[\lambda\ A\ M]_\xi & := & [A]_\xi\ \&\ [\![A]\!]_\xi \in type_\xi\ \&\ [M]_\xi \\
[\![\lambda\ A\ M]\!]_\xi\,(x,\rho) & := & \\
\multicolumn{3}{l}{(f \in \Sigma y \mathcal{C}\,(y,x)) \mapsto \Big(a \in [\![A]\!]_\xi\,(x,\rho\cdot f)\,(id_y)\Big) \mapsto [\![M]\!]_{\xi,[\![A]\!]_\xi}\,(y,(\rho\cdot f,a))}
\end{array}$$

$$\begin{array}{lll}
[M\ N]_\xi & := & [M]_\xi\ \&\ [N]_\xi\ \&\ \forall x,\rho,\ id_x \in dom\Big([\![M]\!]_\xi\,(x,\rho)\Big) \\
 & & \forall x,\rho,\ [\![N]\!]_\xi\,(x,\rho) \in dom\Big([\![M]\!]_\xi\,(x,\rho)\,(id_x)\Big) \\
[\![M\ N]\!]_\xi\,(x,\rho) & := & [\![M]\!]_\xi\,(x,\rho)\,(id_x)\Big([\![N]\!]_\xi\,(x,\rho)\Big)
\end{array}$$

Besides the indirection from the definition of $type_\xi$, the natural definitions work for Σ-types and the associated operations.

$$\begin{array}{lll}
[\Sigma\ A\ B]_\xi & := & [A]_\xi\ \&\ [\![A]\!]_\xi \in type_\xi\ \&\ [B]_{\xi,[\![A]\!]_\xi}\ \&\ [\![B]\!]_{\xi,[\![A]\!]_\xi} \in type_{\xi,[\![A]\!]_\xi} \\
[\![\Sigma\ A\ B]\!]_\xi\,(x,\rho)\,(f) & := & \Big(a \in [\![A]\!]_\xi\,(y,\rho\cdot f)\,(id_y)\Big) \times [\![B]\!]_{\xi,[\![A]\!]_\xi}\,(y,(\rho\cdot f,a))\,(id_y) \\
[\![\Sigma\ A\ B]\!]_\xi\,(x,\rho)\,(f\backslash g) & := & \\
 & & (a,m) \mapsto \Big([\![A]\!]_\xi\,(x,\rho)\,(f\backslash g)\,(a)\,,\ [\![M]\!]_{\xi,[\![A]\!]_\xi}\,(x,(\rho,a))\,(f\backslash g)\,(m)\Big)
\end{array}$$

$$\begin{array}{lll}
[M,N]_\xi & := & [M]_\xi\ \&\ [N]_\xi \\
[\![M,N]\!]_\xi\,(x,\rho) & := & \Big([\![M]\!]_\xi\,(x,\rho)\,,[\![N]\!]_\xi\,(x,\rho)\Big)
\end{array}$$

$$\begin{array}{lll}
[M.1]_\xi\,,\,[M.2]_\xi & := & [M]_\xi\ \&\ \forall x,\rho,\ \exists X,Y,\ [\![M]\!]_\xi\,(x,\rho) = (X,Y) \\
[\![M.1]\!]_\xi\,(x,\rho) & := & \pi_1\Big([\![M]\!]_\xi\,(x,\rho)\Big) \\
[\![M.2]\!]_\xi\,(x,\rho) & := & \pi_0\Big([\![M]\!]_\xi\,(x,\rho)\Big)
\end{array}$$

For the regular universes we use the construction of [HS97] to get presheaves and then precompose by the forgetful functor from $(\mathcal{C}/x)^{op}$ to $\mathcal{C}^{op}$ to get a definition that fits into $type_\xi$. The morphisms are then only reindexing. For the proposition universe we will also use the previous lifting but on 2, which is the proposition universe for sets, beside the fact that it is not a Grothendieck universe. Indeed we only want it to classify monomorphisms, that can be seen as fibrations with fibers in 2.

$$\begin{array}{lll}
[\mathcal{U}_l]_\xi\,,\,[prop]_\xi & := & \top \\
[\![\mathcal{U}_l]\!]_\xi\,(x,\rho)\,(f) & := & Cat\,((\mathcal{C}/y)^{op}\,,\mathcal{U}_l) \\
[\![\mathcal{U}_l]\!]_\xi\,(x,\rho)\,(f\backslash g) & := & g^* \\
[\![\Omega]\!]_\xi\,(x,\rho)\,(f) & := & Cat\,((\mathcal{C}/y)^{op}\,,2) \\
[\![\Omega]\!]_\xi\,(x,\rho)\,(f\backslash g) & := & g^*
\end{array}$$

$*$ is to be considered as the unique witness of any true proposition, so it must be 0 at each stages.

$$
\begin{array}{lll}
[*]_\xi & := & \top \\
[\![*]\!]_\xi & := & 0
\end{array}
$$

We define the equality type as 1 at each stage where the two parameter terms are equal and 0 at the others. This way we get a type in Ω. The morphisms must be the identity morphisms because a semantic proposition is either 0 or 1 at each stages.

$$
\begin{array}{lll}
[M \equiv_A N]_\xi & := & [A]_\xi \ \& \ [\![A]\!]_\xi \ \in type_\xi \\
 & & [M]_\xi \ \& \ [\![M]\!]_\xi \in term_{\xi;[\![A]\!]_\xi} \ \& \ [N]_\xi \ \& \ [\![N]\!]_\xi \in term_{\xi;[\![A]\!]_\xi} \\
[\![M \equiv_A N]\!]_\xi (x,\rho)(f) & := & \left\{ s \mid s = 0 \ \& \ [\![M]\!]_\xi (y, \rho\cdot f) = [\![N]\!]_\xi (y,\rho\cdot f) \right\} \\
[\![M \equiv_A N]\!]_\xi (x,\rho)(f \backslash g) & := & s \mapsto s
\end{array}
$$

We define squashed type likewise.

$$
\begin{array}{lll}
[\|A\|]_\xi & := & [A]_\xi \ \& \ [\![A]\!]_\xi \in type_\xi \\
[\![\|A\|]\!]_\xi (x,\rho)(f) & := & \left\{ s \mid s = 0 \ \& \ \exists X,\ X \in [\![A]\!]_\xi (y, \rho\cdot f) \right\} \\
[\![\|A\|]\!]_\xi (x,\rho)(f \backslash g) & := & s \mapsto s
\end{array}
$$

The semantic of $|M|$ is basically the same as $*$, as propositions have at most one inhabitant. The distinction comes from the fact that we ask more to define $|M|$, we want M to be also defined.

$$
\begin{array}{lll}
[|M|]_\xi & := & [M]_\xi \\
[\![|M|]\!]_\xi & := & [\![*]\!]_\xi
\end{array}
$$

The elimination principle for squashed types states that any constant function $f : A \to B$ factors through a (unique) function $\langle f\rangle : \|A\| \to B$ such that $f = \langle f\rangle \circ |{-}|$. The semantic is then to take the unique element of the codomain whenever there is an element in the domain. As we ask the parameter to be a constant function the union of the codomain is a canonical way of recovering it.

$$
\begin{array}{lll}
[\langle M\rangle]_\xi & := & [N]_\xi \ \& \ \forall x,y,\rho, f : y \to x,\ f \in dom\left([\![\langle M\rangle]\!]_\xi (x,\rho)\right) \\
 & & \forall a,b \in dom\left([\![\langle M\rangle]\!]_\xi (x,\rho)(f)\right), \\
 & & [\![\langle M\rangle]\!]_\xi (x,\rho)(f)(a) = [\![\langle M\rangle]\!]_\xi (x,\rho)(f)(b) \\
[\![\langle M\rangle]\!]_\xi (x,\rho) & := & \\
\multicolumn{3}{l}{(f \in \Sigma y \mathcal{C}(y,x)) \mapsto \left(\_ \in dom\left([\![M]\!]_\xi (x,\rho)(f)\right)\right) \mapsto \bigcup_{x \in cod\left([\![M]\!]_\xi (x,\rho)(f)\right)} x}
\end{array}
$$

Weakening and substitution As a first step toward showing the fitness of our semantic we show that it is well-behaved with regard to weakening and substitution.

First we prove the *weakening lemma* which states that the semantic of a term is preserved by adding an unused variable to the context. We express the fact that a context

ξ extends a context ξ' by adding a variable at position k by the equation $\xi' = \hat{\pi}_k(\xi)$. Let be $n, k \in \mathbb{N}$ with $k \leq n$, $M \in term_n$ and $\xi \in \widehat{\mathcal{C}}_{n+1}$. We prove by induction on M that

$$[M]_{\hat{\pi}_k(\xi)} \iff [\uparrow_k M]_\xi$$

$$\forall x \in \mathcal{C},\, \forall \rho \in \xi_x,\, [\![M]\!]_{\hat{\pi}_k(\xi)}(x, \hat{\pi}(\rho)) = [\![\uparrow_k M]\!]_\xi(x, \rho)$$

For the most of the cases the induction is straightforward. The only difficulties come from the binders, so here we will focus on the Π-type former. The full proof can be found in annex A.1.

Let $M = \Pi\ A\ B$. It follows from the induction hypothesis on A that $[A]_{\hat{\pi}(\xi)} \iff [\uparrow_k A]_\xi$ and that $[\![A]\!]_{\hat{\pi}_k(\xi)} \in type_{\hat{\pi}_k(\xi)} \iff [\![\uparrow_k A]\!]_\xi \in type_\xi$. Next we have to prove that

$$[B]_{\hat{\pi}_k(\xi), [\![A]\!]_{\hat{\pi}_k(\xi)}} \iff [\uparrow_k B]_{\xi, [\![\uparrow_k A]\!]_\xi}$$

But we known from the induction hypothesis on B that

$$[B]_{\hat{\pi}_{k+1}\left(\xi, [\![\uparrow_k A]\!]_\xi\right)} \iff [\uparrow_k B]_{\xi, [\![\uparrow_k A]\!]_\xi}$$

It is then enough to prove that

$$\hat{\pi}_k(\xi), [\![A]\!]_{\hat{\pi}_k(\xi)} = \hat{\pi}_{k+1}\left(\xi, [\![\uparrow_k A]\!]_\xi\right)$$

Let be $(\rho, a) \in \hat{\pi}_k(\xi), [\![A]\!]_{\hat{\pi}_k(\xi)}(x)$, with $x \in \mathcal{C}$. There exists $\rho' \in \xi_x$ such that $\rho = \hat{\pi}_k(\rho')$. Then we know from the induction hypothesis that $[\![A]\!]_{\hat{\pi}_k(\xi)}(x, \rho) = [\![\uparrow_k A]\!]_\xi(x, \rho')$, so that $a \in [\![\uparrow_k A]\!]_\xi(x, \rho')(id_x)$. It follows that $(\rho', a) \in \xi, [\![\uparrow_k A]\!]_\xi(x)$ and by computations on the projection that $(\rho, a) \in \hat{\pi}_{k+1}\left(\xi, [\![\uparrow_k A]\!]_\xi\right)(x)$. This proves that for each $x \in \mathcal{C}$ we have $\hat{\pi}_k(\xi), [\![A]\!]_{\hat{\pi}_k(\xi)}(x) \subseteq \hat{\pi}_{k+1}\left(\xi, [\![\uparrow_k A]\!]_\xi\right)(x)$, but all the proof can be fully inverted to get the opposite inclusion, so that these two sets are in fact equal. To conclude that the two presheaves are the same we also need to prove the equality on arrows, but the proof would be the same as for the objects.

The rest of the proof is straightforward rewriting using the hypothesis.

The *substitution lemma* states likewise that semantic is preserved by substitution. Let be $n, k \in \mathbb{N}$, $S \in term_n$ and $\xi \in \widehat{\mathcal{C}}_{n+1+k}$ such that $[S]_{\pi_{->k}(\xi)}$ and

$$\forall x \in \mathcal{C},\, \forall \rho \in \xi_x,\, \pi_k(\rho) = [\![S]\!]_{\pi_{->k}(\xi)}(x, \pi_{->k}(\rho))$$

Then for all $M \in term_{n+1+k}$ we have

$$[M]_\xi \iff [M[k \setminus S]]_{\hat{\pi}_k(\xi)}$$

$$\forall x \in \mathcal{C},\, \forall \rho \in \xi_x,\, [\![M]\!]_\xi(x, \rho) = [\![M[k \setminus S]]\!]_{\hat{\pi}_k(\xi)}(x, \hat{\pi}_k(\rho))$$

The proof of the substitution lemma has the same structure as the substitution lemma and uses the same arguments. Nevertheless we have to use the weakening lemma in the variable case. The details can be found in annex A.2.

Statement's interpretation Let be $\mathcal{J} \in judgement_n$, $n \in \mathbb{N}$. We give the semantic of $\mathcal{J}$ under a semantic context $\xi \in \widehat{\mathcal{C}_n}$ as a predicate $[\mathcal{J}]_\xi$.

$$\begin{array}{lll}
[\mathrm{ok}]_\xi & := & \top \\
[A\ type]_\xi & := & [A]_\xi \ \& \ [\![A]\!]_\xi \in type_\xi \\
[A = B\ type]_\xi & := & [A\ type]_\xi \ \& \ [B\ type]_\xi \ \& \ [\![A]\!]_\xi = [\![B]\!]_\xi \\
[M : A]_\xi & := & [M]_\xi \ \& \ [A\ type]_\xi \ \& \ [\![M]\!]_\xi \in term_{\xi,[\![A]\!]_\xi} \\
[M = N : A]_\xi & := & [M : A]_\xi \ \& \ [N : A] \ \& \ [\![M]\!]_\xi = [\![N]\!]_\xi
\end{array}$$

We then define by induction the semantic $[\![\Gamma]\!] \in \widehat{\mathcal{C}_n}$ of a context $\Gamma \in context_n$ and the conditions under which this semantic is defined as a predicate $[\Gamma]$.

$$\begin{array}{lll}
[\epsilon] & := & \top \\
[\Gamma, A] & := & [\Gamma] \ \& \ [A]_{[\![\Gamma]\!]} \ \& \ [\![A]\!]_{[\![\Gamma]\!]} \in type_{[\![\Gamma]\!]} \\
[\![\epsilon]\!] & := & 1 \\
[\![\Gamma, A]\!] & := & [\![\Gamma]\!], [\![A]\!]_{[\![\Gamma]\!]}
\end{array}$$

Finally we can define the semantic of a statement $\mathcal{S} = \Gamma \vdash \mathcal{J}$ as the predicate

$$[\mathcal{S}] \ := \ [\Gamma] \ \& \ [J]_{[\![\Gamma]\!]}$$

1.3 Derivation rules

The final step of the process of justifying our type theory is to prove the derivation rule themselves. By proving a derivation rule we mean to prove that if the semantic of the rule's hypothesis hold then the semantic of its conclusion does. Most of the rules we prove are usual for a Martin-Löf type theory, so we only present here the specificities of our theory. The full list of rules can be found in annex A.3 along their demonstration.

We prove that we have universies *à la* Russel, that is to say that an element of a universe does not need an operator translating it into a type.

$$\frac{\Gamma \vdash A : \mathcal{U}_l}{\Gamma \vdash A\ type} \qquad \frac{\Gamma \vdash A : \Omega}{\Gamma \vdash A\ type}$$

Moreover the universe hierarchy is cumulative. This means that elements of one universe are member of the ones above.

$$\frac{\Gamma \vdash A : \mathcal{U}_l}{\Gamma \vdash A : \mathcal{U}_{l+1}} \qquad \frac{\Gamma \vdash A : \Omega}{\Gamma \vdash A : \mathcal{U}_0}$$

We have strict propositions in the sens that a proposition is at most uniquely inhabited by $*$. This has the practical consequence that all proofs are equal and we don't need to care about them.

$$\frac{\Gamma \vdash A : \Omega \qquad \Gamma \vdash M : A}{\Gamma \vdash M = * : A}$$

Moreover we also have that two propositions with the same proof strength are equal.

$$\frac{\Gamma \vdash A : \Omega \qquad \Gamma \vdash B : \Omega \qquad \Gamma, A \vdash M : B \qquad \Gamma, B \vdash N : A}{\Gamma \vdash A = B : \Omega}$$

It is enough to build most propositions to have the propositional truncation. Hence we define for each type A a proposition $\|A\|$ which holds whenever A is inhabited.

$$\frac{\Gamma \vdash A \ type}{\Gamma \vdash \|A\| : \Omega} \quad \frac{\Gamma \vdash M \in A}{\Gamma \vdash |M| : \|A\|}$$

Moreover we can eliminate squashed type thanks to the fact that constant functions factor through their squashed domain.

$$\frac{\Gamma \vdash N : A \to B \qquad \Gamma, A, A \vdash N \ var_1 = N \ var_0 : B}{\Gamma \vdash \langle N \rangle : \|A\| \to B}$$

$$\frac{\Gamma \vdash N : A \to B \qquad \Gamma, A, A \vdash N \ var_1 = N \ var_0 : B \qquad \Gamma \vdash M : A}{\Gamma \vdash \langle N \rangle \ |M| = N \ M : B}$$

Finally we define the equality type to be extensional.

$$\frac{\Gamma \vdash T : M \equiv_A N}{\Gamma \vdash M = N : A}$$

1.4 Discussion

As there is no issue in translating usual variable notation into DeBruijn indexes we will use from now the common syntax.

In order to be useful our theory would need the addition of several “ regular ” data types. In is important to note that the new symbol introduced to interpret additions to the theory would require one the update the proofs of weakening and substitution, but symbols for constants come with no effort. In particular there is no issue in adding a boolean type $\mathbb{B}$ and a natural number type $\mathbb{N}$. To do so we interpret them as constant presheaves, defining their operations is them straightforward.

A more interesting addition would be the comprehension subtype $\{x : A \mid \varphi(x)\}$, where $\varphi : A \to \Omega$, used in [OP17]. It could be built as different encoding of $\Sigma(x : A)\,\varphi(x)$ which take advantage of the proof's irrelevance. This would allow a faithful translation of [OP17] in our setting but one could argue that instead the construction $\Sigma(x : A)\,\varphi(x)$ could be used directly. In fact we use a quite different formulation of propositions, as defining $\Sigma(x : A)\,\varphi(x)$ would require a comprehension subtype in [OP17].

Example of interpretation We unfold the interpretation of the type

$$\Pi\,(\varphi : \mathbb{F})\,\Pi\,(u : \varphi \to A)\,\Sigma\,(x : A)\,\Pi\,(y : \varphi)\,x \equiv_A u\ y$$

where $\mathbb{F}$ is a sub-presheaf of Ω and A is a free variable of type $\mathcal{U}_n$. Having a term of this type means that for any object $o \in \mathcal{C}$, any type $a : A \to \mathsf{y}o$, any proposition $\varphi : \mathsf{y}o \to \mathbb{F}$ and any morphism $u : [\varphi] \to A$ such that $a \circ u = \varphi$ (with y being the Yoneda embedding and $[\varphi] \to \mathsf{y}o$ being the pullback of φ against the subobject classifier) there exists an section $x : \mathsf{y}o \to A$ of $a : A \to \mathsf{y}o$ such that u factors through x.

This interpretation as an alternative to Kripke-Joyal semantics as a tool to enable the use of logics in the study of categories. The main difference we see in the example between the two semantics is that our semantic does not fully speak in terms of generalized elements but in terms of elements with a representable codomain. But as any object in a presheaf category is a colimit (generalized sum) of representable object we think that we may recover the full Kripke-Joyal from this and even extend it, which would make our interpretation is generalization of Kripke-Joyal in presheaves. Nevertheless we never focused on this particular point.

2 Extension structure and the boolean stack model

In this section we will use the internal language of presheaves to study a notion of *extension structure*. Its aim is allow one to extend a partial element on a proposition φ, ie. a member of $\varphi \to A$ for some type A, into a global element of A. The inspiration for this construction is the Bousfield localization [Hir09], which is the process of adding trivial cofibration to a Quillen model structure. In our attempt of internalizing this notion we call *trivial* the propositions along which we will allow partial elements to be extended.

First we will present this as a way to translate a theory where selected propositions are true to an other where they may not be. Then we will show how to adapt the construction to an internal language that satisfies the axioms of [OP17] in order to get a model of the univalence axiom. Thanks to this we will exhibit a model of both univalence and Brouwer's fan theorem.

The justification of Brouwer's fan theorem from a boolean stack model is an immediate adaptation of a proof of my supervisor for a boolean sheaf model. The "stackification" of booleans and natural numbers are personal. Besides that all the work done here has been suggested by Coquand, the proofs are personal.

2.1 Extension structure in dependent type theory

We give ourselves a dependent type theory as described in the first section. We will annotate the sequents of this theory with a subscript 1, hence $\Gamma \vdash \mathcal{S}$ will become $\Gamma \vdash_1 \mathcal{S}$. By extension we will name this theory $\vdash_1$. We assume the existence of a subtype $\mathbb{T}$ of Ω. Our aim will be to translate into $\vdash_1$ a dependent type theory $\vdash_2$ in which for all type

A there exists a canonical way of extending an element of $\varphi \to A$ into an element of A, where $\varphi : \mathbb{T}$, such that an element and its extension are equal where φ holds.

We will ask that $\mathbb{T}$ satisfies the " strictness " axiom from [OP17] for each universe level, where $\cong$ is the type of isomorphisms, ie.

$$A \cong B := \Sigma\,(f : A \;\to B)\,\Sigma\,(g : B \to A)\, f \circ g = id_B \wedge g \circ f = id_A$$

$$\begin{array}{ll} \vdash_1 \texttt{strict}_l : & \Pi\,(\varphi : \mathbb{T})\,\Pi\,(A : \varphi \to \mathcal{U}_l)\,\Pi\,(B : \mathcal{U}_l)\,\Pi\,(s : \Pi\,\varphi\,(A * \cong B)) \\ & \Sigma\,(B' : \mathcal{U}_l)\,\Sigma\,(s' : B' \cong B)\,\Pi\,\varphi\,(A * = B' \wedge s * = s') \end{array}$$

This axioms always holds for presheaf models when we work in a classical metatheory, but does not in a constructive one. In presheaves it corresponds intuitively to saying that the monomorphisms classified by $\mathbb{T}$ have decidable image at each stage of the base category. The reciprocal is proven true in [OP17].

For each type A in $\vdash_1$ we define an extension structure $\texttt{Ext}_A$ to be an operation $\rho : \Pi\,(\varphi : \mathbb{T})\,(a : \Pi\;\varphi\,A)\,A$, called the extension, such that for each $\varphi : \mathbb{T}$ we have that $\rho\;\varphi$ is a right inverse of the restriction along φ, ie. a right inverse of $\lambda\,(a : A)\,\lambda\,(\_ : \varphi)\,a$. In practice we will explain this condition is a different but equivalent way.

$$\texttt{Ext}_A := \Pi\,(\varphi : \mathbb{T})\,\Sigma\,(\rho : \Pi\;\varphi\,A \to A)\,(\varphi \to \Pi\,(a : \Pi\;\varphi\,A)\;\;\rho\;a = a*)$$

We will then translate a type A in $\vdash_2$ as a type A' in $\vdash_1$ equipped with an extension structure $p : \texttt{Ext}_{A'}$. We prove by induction the translation of the type formers.

We start with Π-types. Let A be a type (that may not have extension structures) and a type B on A such that for each $a : A$ we have an extension structure $q\;a$ for $B\;a$. Then we have an extension structure for $\Pi\;A\;B$. Indeed, let be $\varphi : \mathbb{T}$ and $f : \varphi \to \Pi\;A\;B$. We define $p'\;f\;a : B\;a$ to be the extension of $\lambda\;\varphi\;\;(f\;*\;a) : \varphi \to B\;a$. This define a function $p' : (\varphi \to \Pi\;A\;B) \to \Pi\;A\;B$. Now we assume that φ holds and check the constraint.

$$\begin{array}{lll} p'\;f & = & \lambda\,(a : A)\;\;q\;a\;(\lambda\;\varphi\;(f\;*\;a)) \\ & = & \lambda\,(a : A)\;\;f\;*\;a \\ & = & f\;* \end{array}$$

Note that we don't need the domain of a Π-type to have an extension structure. For the Σ-type $\Sigma\;A\;B$ we will need A to have an extension structure p. Let be $\varphi : \mathbb{T}$ and $t : \varphi \to \Sigma\;A\;B$. We first extend the first projection of t into an element $a : A$. Because $\varphi \to t\;*\;.1 = a$ we can extend the second projection of t thanks to $q\;a$ into $b : B\;a$. That way we build the extension of a Σ-type, the constraint being straightforward to check.

We lift an element A of a universe, ie. a type, on φ thanks to $\Pi\;\varphi\;A$. More precisely, if A is an element of a predicative universe we use the strictness axiom. If A is a proposition we have to squash $\Pi\;\varphi\;A$ to get a proposition which is definitionally equal to A under φ. The details are in the annex B.1. We do not care about the lifting of $\mathbb{T}$ itself, but it would require $\mathbb{T}$ to be stable under dependent product to be able to do so.

We are not able to lift equality types with the current definition of an extension structure, nor we are able to lift propositional truncation. We can prove that being able to extend an equality between elements of a type A, itself equipped with an extension structure, is equivalent to saying that the extension on A is an isomorphism. But if we ask extensions to be isomorphisms then we are no longer able to lift the predicative universes. Anyway HOTT and CTT provide for these types replacements that we will be able to lift. The intuition there is that these theories allow one to work nicely with non-strict constraints.

An other issue with this construction is that we are not able to lift definitional booleans and natural numbers. The problem comes from that despite being able to describe the lifted type (externally using a " sheafification ", see [MM94]) we are not able in general to lift the corresponding strict equalities.

2.2 Extension structure in cubical type theory

We adapt this extension structure to cubical type theory, in order to have a type theory with both univalence and extensions for some propositions. We suppose that $\vdash_1$ satisfies the axioms in [OP17].

Roughly speaking, their axioms suppose the existence of two types $\mathbb{I}$ and $\mathbb{F}$ such that $\mathbb{I}$ is a path connection algebra (a De Morgan algebra without negation) with two distinct endpoints $\mathtt{0}$ and $\mathtt{1}$ that are connected : if a proposition on $\mathbb{I}$ is decidable then it has the same truth value on both endpoints. $\mathbb{F}$ is supposed to be a subtype of Ω stable by union, dependent product, quantification over $\mathbb{I}$ and equality with endpoints of $\mathbb{I}$. Finally the original strictness axiom from [OP17] has its first quantification over $\mathbb{F}$, so here we state it that way.

$$\begin{aligned} \vdash_1 \mathtt{strict}_l : \quad & \Pi\,(\varphi : \mathbb{F})\,\Pi\,(A : \varphi \to \mathcal{U}_l)\,\Pi\,(B : \mathcal{U}_l)\,\Pi\,(s : \Pi\,\varphi\,(A\,* \cong B)) \\ & \Sigma\,(B' : \mathcal{U}_l)\,\Sigma\,(s' : B' \cong B)\,\Pi\,\varphi\,(A\,* = B' \wedge s\,* = s') \end{aligned}$$

The composition structure studied in [OP17] can then be thought as an extension structure for propositions of the form $(i = x) \vee \varphi$ where $i : \mathbb{I}$, x is $\mathtt{0}$ or $\mathtt{1}$ and $\varphi : \mathbb{F}$ does not refer to i. The initial purpose of the extension structure we present in this report was to generalize the propositions under which we can compose to those of the form $\gamma \vee \varphi$ where $\gamma : \mathbb{T}$ and $\varphi : \mathbb{F}$. Nevertheless it seems more natural to add it as a different structure, as it still makes sens outside of models of cubical type theory.

We recover the strictness axiom for $\mathbb{T}$ by asking that $\mathbb{T}$ is a subtype of $\mathbb{F}$. We also assume that $\mathbb{T}$ is stable by union with elements of $\mathbb{F}$.

We will interpret a cubical type theory $\vdash_2$ directly in $\widehat{\mathcal{C}}$ and prove its correctness by using the correspondence between types in $\vdash_2$ and types in $\vdash_1$ equipped with a composition structure of [OP17] and an extension structure. We don't proceed by translation of $\vdash_2$ into $\vdash_1$ because of the negative result that can be found in [LOPS18]. This paper also shows a way to modify our $\vdash_1$ so that we can retrieve the translation at the cost of $\vdash_1$ becoming non-standard, but there is other ways of constructing such universes as done in [CCHM16]. Because the specific problem exposed by [LOPS18] lies in the translation of predicative universes we will not assume that we have an internalization of them in $\vdash_1$. Instead we will show that we can lift both an element of a predicative universe of $\vdash_1$ and its composition structure and rely on the fact that we can built universes for such types in $\vdash_2$.

The extension structures built in the non univalent case are still valid, so we focus here on the additional " unusual " types we want to lift here. Within these types is the type $A : \mathbb{I} \to \mathcal{U}, a : A\ \mathtt{0}, b : A\ \mathtt{1} \vdash_1 \mathit{Path}\ A\ a\ b := \Sigma\,(p : \Pi\ \mathbb{I}\ A)\ \ p\ \mathtt{0} = a \wedge p\ \mathtt{1} = b$ of paths between elements of type on $\mathbb{I}$. We cannot lift it the " trivial " way as we know we cannot lift equalities. Instead we would like to be able to construct an element p' on the extent $\psi := \varphi \vee (i = \mathtt{0}) \vee (i = \mathtt{1})$ and under an abstraction over $i : \mathbb{I}$, such that

$$\begin{array}{rcl} \varphi & \to & p' = p * i \\ i = \mathtt{0} & \to & p' = a \\ i = \mathtt{1} & \to & p' = b \end{array}$$

Because of the constraints on p such definitions would agree on the intersections of the three extents. For instance, when both φ and $i = \mathtt{0}$ hold we have $p * i = a$. The extension of p' over ψ would then give us an element of $\Pi\ \mathbb{I}\ A$ which is indeed a path between a and b. This kind of definitions by case analysis over propositions are called *systems* in [CCHM16]. In order to build the system above we first define the function f from the disjoint union $\varphi + (i = \mathtt{0}) + (i = \mathtt{1})$ that assigns $p * i$ to φ, a to $i = \mathtt{0}$ and b to $i = \mathtt{1}$ and because this function is constant it factors through the (logical) disjunction and we recover its codomain as the system we want.

Thanks to this use of systems and the fact that extension structures are preserved by isomorphisms we then prove an extension structure for the glue types and composition structure from [OP17]. This complete the justification of an extension structure for cubical type theory, the details of the proof are in the annex B.2.

2.3 The boolean stack model

We call a *cube category* a category $\square$ with finite products equipped with an object $\mathtt{i}$ that has the structure of a non-trivial connection algebra (so that the object of $\widehat{\square}$ represented by $\mathtt{i}$ satisfy the axioms of $\mathbb{I}$) such that for all objects $c \in \square$, $\square(\mathtt{c}, \mathtt{i})$ has decidable equality. The usual cube category from [CCHM16] is an instance of this definition. If we take $\mathcal{A}$ to be a category with finite products, then the product category $\square \times \mathcal{A}$ is also a cube category with $\mathtt{i}_{\mathcal{A}} := (\mathtt{i}, 1_{\mathcal{A}})$, where $1_{\mathcal{A}}$ is the terminal object (or 0-product) of $\mathcal{A}$. According to corollary 8.5 from [OP17] the internal language of a cube category satisfies the axioms needed to model CTT.

Let be $\mathcal{B}$ the category of non-degenerate decidable boolean algebras. Its opposite category $\mathcal{B}^{op}$ has all finite products so $\mathcal{C} := \square \times \mathcal{B}^{op}$ is a cube category. The category $\mathcal{B}^{op}$ corresponds through Stone duality to a sub-category of the category of topological spaces : the non-empty compact totally disconnected Hausdorff spaces with an additional decidability condition. The set of elements of a boolean algebra is then in bijection with the both open and closed sets of the corresponding space.

What we will want to describe is geometrically the types whose terms at some boolean algebra can be recovered from a finite partition of that algebra (seen as a space). That is to say that if we have a term of that type for each element of the partition then we can recover a term globally defined on the whole boolean algebra. We express this condition on the types we are interested in as an extension structure. If we describe $\mathbb{T}$ as a presheaf this would mean that at each stages $(I, B) \in \mathcal{C}$ if $\varphi \in \mathbb{T}(I, B)$ then $\varphi \in \mathbb{F}(I, B)$ and there exists a finite partition $B_1, \ldots, B_n$ of B such that for each $i \in \{1, \ldots, n\}$ the restriction $\varphi|_{B_i} \in \mathbb{T}(I, B_i)$ is true. Here true means that the presheaf in $\widehat{\mathcal{C}/(I, B_i)}$ induced by $\varphi|_{B_i}$ is constant at 1. We say that the partition $B_1, \ldots, B_n$ makes φ true. When we then unfold the semantic of $\vdash_1$ this means that having an element of $\varphi \to A$ at stage (I, B) under a parameter ρ from the context implies that for each $i \in \{1, \ldots, n\}$ we have an element of A at stage (I, B_i) under the parameter $\rho|_{B_i}$. Hence extending along $\varphi \to A$ is recovering an element from a partition, although the whole operation is a bit more general than that in order to have $\mathbb{T}$ stable by unions with elements of $\mathbb{F}$.

Because we asked our boolean algebras to be decidable, the presheaf over $\mathcal{C}/(I, B)$ induced by such a partition is a decidable sub-object of the true presheaf. This means that if we take $\mathbb{F}$ to be the presheaf of decidable sub-presheaves of true, as proposed in [OP17], then we can take elements of $\mathbb{T}$ in $\mathbb{F}$ and keep the propositions induced by the finite partitions. Formally, we will take $\mathbb{T}$ the presheaf of decidable sub-presheaves of true such that, for each stage (I, B) if $\varphi \in \mathbb{T}(I, B)$ then there exists a finite partition of B which makes φ true. This choice of $\mathbb{T}$ satisfies by construction the axioms needed to get an extension structure in a cubical type theory.

Booleans and natural numbers We show that in this particular model of CTT with extension structure we can externally describe the type of booleans and natural numbers. More generally we will explain how we can lift a set S with decidable equality and its operations to $\vdash_2$. First we define a presheaf P_S by

$$P_S(I, B) = \{f : S \to B \mid \begin{array}{l} \{x \mid f(x) \neq 0\} \in \mathcal{P}_{fin}(S) \\ \forall x \neq y \in S,\ f(x) \wedge f(y) = 0 \\ \bigvee_{x \in S} f(x) = 1 \end{array} \}$$

$P_S(u : J \to I, v^{op} : C \to B) : P_S(I, B) \to P_S(J, C)$ is defined as the postcomposition by v. The invariants of P_S are indeed preserved because v is morphism of boolean algebras.

Assume an object $(I, B) \in \mathcal{C}$, a proposition $\varphi \in \mathbb{T}$ and an element e of P_S defined on the extent of φ. Because of the definition of φ we have a partition $B_1, \ldots, B_n$ which makes φ true. In particular e is defined for each B_i. Moreover, if we have an element $b_i \in B_i$ for each $i \in \{1, \ldots, n\}$ we recover the unique element $b \in B$ which restrict to b_i on B_i for each i by the universal property of the non-empty products in $\mathcal{B}$ (replacing S with the free boolean algebra on S). That way we get an element of P_S at stage (I, B). Let now be $(u : J \to I, v^{op} : C \to B)$ such that $\varphi(u, v^{op}) = 1$, when looking at φ as a presheaf on $\mathcal{C}/(I, B)$. The image of $B_1, \ldots, B_n$ through v gives a partition of C which makes $\varphi \cdot (u, v^{op}) \in \mathbb{T}(J, C)$ true, so that we can use the same argument to describe the value of e at (J, C) (by assumption e is already defined here). By construction of the partition on C we will then have that the element of $P_S(I, B)$ will indeed restrict the element of $P_S(J, C)$ we just built. We successfully constructed the extension structure of P_S.

Because $P_S(I, B) = P_S(J, B)$ for each $I, J \in \square$ and $B \in \mathcal{B}$ an explicit calculation of the path types over P_S shows that there are no non trivial paths between elements of P_S. This and the extension of P_S entails that P_S can be lifted to $\vdash_2$. Now we have to lift the operations on S. For instance, let's take $S = \mathbb{N}$. The zero constructor is defined to be $(0 \mapsto 1, \_ \mapsto 0)$ at each stage (I, B) and the successor function is $f \mapsto (0 \mapsto 0, (s+1) \mapsto f(s))$. We define the recursor

$$A : \mathbb{N} \to \mathcal{U}_l,\ i : A\ 0,\ f : \Pi\,(n : \mathbb{N})\,(A\ n \to A\ \ (n+1))\,,\ n : \mathbb{N} \vdash_2 \texttt{natrec}\ A\ i\ f\ n : A\ n$$

at stage (I, B) by first restricting ourselves to the partition of B entailed by the domain of (the semantic of) n. On each element of this partition n is a regular natural number so we can do the recursion. If we consider the domain of n as an element φ of $\mathbb{T}(I, B)$, which it is by definition of $\mathbb{T}$, then we just built an element of $\varphi \to A\ n$. We don't have to check whether the elements on the partition are compatible because there is no intersection of disjoint subspaces in $\mathcal{B}^{op}$, indeed we removed the degenerate boolean algebra from $\mathcal{B}$. We then use the extension structure of $A\ n$ is $\vdash_1$ to extend this element to $A\ n$. The definitional equalities for the recursor in $\mathbb{N}$ is then entailed by the constraint on extension structures.

We do the same for the booleans in $\vdash_2$.

Fan theorem Let $\mathcal{U}$ be some universe. We prove that the fan theorem holds in the boolean stack model :

$$T : 2^{\mathbb{N}} \times \mathbb{N} \to \mathcal{U},\ p : \Pi\left(\alpha : 2^{\mathbb{N}}\right)\Sigma\,(n : \mathbb{N})\,T\,(\alpha, n)$$
$$\vdash_2 \texttt{fan}\ T\ p : \Sigma\,(M : \mathbb{N})\,\Pi\left(\alpha : 2^{\mathbb{N}}\right)\Sigma\,(n : \mathbb{N})\,n \leq M \wedge T\,(\alpha, n)$$

First we can see that the interpretation of the type $\mathbb{B}$ of the booleans is isomorphic to the presheaf $\pi_0(I, B) = B$, because an element $f : 2 \to B$ of $\mathbb{B}(I, B)$ is uniquely determined by the image of 1 (or 0, it doesn't matter which one). Then the semantic of $2^{\mathbb{N}}$ is isomorphic at stage (I, B) to the functions $\mathbb{N} \to B$. This comes from the fact that we do not need to lift into $\vdash_2$ the domain on a Π-type, and then that the functions with lifted domain are isomorphic with the function with unlifted domain. If we take the object $C \in \mathcal{B}$ to be the boolean algebra generated by countably many elements then we have that $2^{\mathbb{N}}$ is represented by $A := 1_\square \times C$, that is to say that $2^{\mathbb{N}}$ is isomorphic with the morphisms to A in $\widehat{\mathcal{C}}$.

The hypothesis of the fan theorem then means that for each morphisms $f : B' \to B$ and $\alpha : B' \to A$ in $\mathcal{C}$ there exists a partition B_i of B' and for each i a natural number n_i such that $T(\alpha, n_i)$ is inhabited at B_i for every i. To prove the theorem we then have to choose an upper bond of each n_i that can only depend on B, and prove that this choice is natural.

To that extent we take B' to be the product $B \times A$ (which exists because $\mathcal{C}$ is a cube category and thus has finite products) and f and α to be the projections. This gives us a partition of $B \times A$ and natural numbers which provide inhabitants for T. But because of the universal property of the products, any two arrows $f : B' \to B$ and $\alpha : B' \to A$ factors through $B \times A$ and the corresponding projections. This means that the result at $B \times A$ will restrict to any arrows under B, so we can choose at stage B a partition and natural numbers that we will be greater than the ones at $B \times A$.

It remains to prove that we can make this choice into a natural transformation. To that extent we will rely of an explicit description of the coproduct of boolean algebras, which becomes the product in $\mathcal{C}$. An element of $B \times A$ is made of two partition $\mathcal{B}$ and $\mathcal{A}$ of respectively B and A and a coloring $c : \mathcal{B} \times \mathcal{A} \to 2$ for all $x \neq y \in \mathcal{B}$ we have $c(x, -) \neq c(y, -)$ and for all $x \neq y \in \mathcal{A}$, $c(-, x) \neq (-, y)$. The inverse image of 1 by c is then a finite formal sum of pairs of elements (x, y) where $x \in B$ and $y \in A$. The operations are taken pointwise and we take care of collapsing the partitions $\mathcal{B}$ and $\mathcal{A}$ to preserve the invariant of the coloring. A partition with natural numbers of $B \times A$ is then a function c like before but with codomain any finite subset of $\mathbb{N}$. Our choice of upper bound that does depend on $\mathcal{A}$ is then to take the maximum each line $c(x, -)$ along $x \in (B)$ in order to get a partition with natural numbers on $\mathcal{B}$. Such choice is natural because the maximum of lines along $\mathcal{B}$ in $\mathcal{B} \times \mathcal{A}$ does not change along restrictions of the form $f \times id_{\mathcal{A}} : B' \times A \to B \times A$ with $f : B' \to B$.

Conclusion

This internship report aimed at proving the well-definedness and the usefulness of internal languages in the study of models of dependent type theory. The ease with which we prove the compatibility between univalence and the fan theorem should be a testimony of that simplicity. In fact I spent most of my time during that internship gathering the knowledge required to work about these subjects and trying out definitions, as the proofs themselves are often immediate and straightforward.

We think that the compatibility result found in this internship is worth a publication by itself, as the fan theorem is the key lemma to enable point-wise topology in a constructive theory. We also believe that we could get an other intuitionistic principle in that setting, the fact that all functions from the real numbers onto themselves are continuous, with little efforts. We hope to find in the future more use of the constructions described here, as they allow the description of more models of dependent type theory with univalence.

On a more personal note this internship has been invaluable in all it allowed me to learn about dependent type theories. Also, I can say from this experience that logic in computer science is the field in which I want to specialize.

References

[Acz98] Peter Aczel. On relating type theories and set theories. In *Types for Proofs and Programs, International Workshop TYPES '98, Kloster Irsee, Germany, March 27-31, 1998, Selected Papers*, pages 1–18, 1998.

[Bar10] Bruno Barras. Sets in coq, coq in sets. *J. Formalized Reasoning*, 3(1):29–48, 2010.

[CCHM16] Cyril Cohen, Thierry Coquand, Simon Huber, and Anders Mörtberg. Cubical type theory: a constructive interpretation of the univalence axiom. *CoRR*, abs/1611.02108, 2016.

[Dyb00] Peter Dybjer. A general formulation of simultaneous inductive-recursive definitions in type theory. *J. Symb. Log.*, 65(2):525–549, 2000.

[GS17] Nicola Gambino and Christian Sattler. The frobenius condition, right properness, and uniform fibrations. 2017.

[Hir09] Philip Hirschhorn. *Model categories and their localizations*, volume 99 of *Mathematical Surveys and Monographs.* American Mathematical Society, 2009.

[HS97] Martin Hofmann and Thomas Streicher. Lifting grothendieck universes, 1997.

[LOPS18] Daniel R. Licata, Ian Orton, Andrew M. Pitts, and Bas Spitters. Internal universes in models of homotopy type theory. *CoRR*, abs/1801.07664, 2018.

[ML79] Per Martin-Löf. Constructive mathematics and computer programming. *Logic, Methodology and Philosophy of Science*, VI:176–184, 1979.

[MM94] Saunders Mac Lane and Ieke Moerdijk. *Sheaves in Geometry and Logic.* Springer-Verlag New York, 1994.

[OP17] Ian Orton and Andrew M. Pitts. Axioms for modelling cubical type theory in a topos. *CoRR*, abs/1712.04864, 2017.

[Str91] Thomas Streicher. *Semantics of Type Theory: Correctness, Completeness, and Independence Results.* Birkhauser Boston Inc., Cambridge, MA, USA, 1991.

[Voe10] Vladimir Voevodsky. Univalent foundations project. *a modified version of an NSF grant application*, page 1–12, October 2010.

A Proofs from section 1

A.1 Semantic weakening

Let be $n, k \in \mathbb{N}$ with $k \leq n$, $N \in term_n$ and $\xi \in \widehat{\mathcal{C}}_{n+1}$. We prove by induction that

$$[M]_{\hat{\pi}_k(\xi)} \iff [\uparrow_k M]_\xi$$

$$\forall x \in \mathcal{C},\ \forall \rho \in \xi_x,\ [\![M]\!]_{\hat{\pi}_k(\xi)}(x, \hat{\pi}_k(\rho)) = [\![\uparrow_k M]\!]_\xi (x, \rho)$$

- $\uparrow_k \ var_i,\ i < k = var_i$

  We first have to prove that $[var_i]_{\hat{\pi}_k(\xi)} \iff [var_i]_\xi$, but both sides are true by definition.

  Let be $x \in \mathcal{C}$ and $\rho \in \xi_x$. We have to prove that $[\![var_i]\!]_{\hat{\pi}_k(\xi)}(x, \hat{\pi}_k(\rho)) = [\![var_i]\!]_\xi (x, \rho)$ that is to say $\rho_i = \rho_i$ because $i < k$.

- $\uparrow_k \ var_i,\ i \geq k = var_{i+1}$

  We first have to prove that $[var_i]_{\hat{\pi}_k(\xi)} \iff [var_{i+1}]_\xi$, but both sides are true by definition.

  Let be $x \in \mathcal{C}$ and $\rho \in \xi_x$. We have to prove that $[\![var_i]\!]_{\hat{\pi}_k(\xi)}(x, \hat{\pi}_k(\rho)) = [\![var_{i+1}]\!]_\xi (x, \rho)$ that is to say $\rho_{i+1} = \rho_{i+1}$ because $i \geq k$.

- $\uparrow_k \ M\ N = (\uparrow_k M)\ (\uparrow_k N)$

  This is straightforward rewriting thanks to the induction hypothesis.

- $\uparrow_k \ \lambda\ A\ M = \lambda\ (\uparrow_k A)\ (\uparrow_{k+1} M)$

  It follows from the induction hypothesis on A that $[A]_{\hat{\pi}_k(\xi)} \iff [\uparrow_k A]_\xi$ and then that $[\![A]\!]_{\hat{\pi}_k(\xi)} \in type_{\hat{\pi}_k(\xi)} \iff [\uparrow_k A]_\xi \in type_\xi$. To prove the equivalence we then have to prove that

  $$[M]_{\hat{\pi}_k(\xi)\cdot[\![A]\!]_{\hat{\pi}_k(\xi)}} \iff [\uparrow_{k+1} M]_{\xi\cdot[\![\uparrow_k A]\!]_\xi}$$

  We know thanks to the induction hypothesis on M that

  $$[M]_{\hat{\pi}_{(k+1)}\left(\xi\cdot[\![\uparrow_k A]\!]_\xi\right)} \iff [\![\uparrow_{k+1} M]\!]_{\xi\cdot[\![\uparrow_k A]\!]_\xi}$$

  Hence we only have to prove that $\hat{\pi}_k(\xi) \cdot [\![A]\!]_{\hat{\pi}_k(\xi)} = \hat{\pi}_{(k+1)}\left(\xi \cdot [\![\uparrow_k A]\!]_\xi\right)$ to get the equivalence we want.

  Let be $(\rho, a) \in \hat{\pi}_k(\xi) \cdot [\![A]\!]_{\hat{\pi}_k(\xi)}(x)$ with $x \in \mathcal{C}$. By definition, $\rho \in \hat{\pi}_k(\xi)_x$ and $a \in [\![A]\!]_{\hat{\pi}_k(\xi)}(x, \rho)(id_x)$. $\rho \in \hat{\pi}_k(\xi)_x$ implies that there exists $\rho' \in \xi_x$ such that $\rho = \hat{\pi}_k(\rho')$. Thanks to the induction hypothesis on A we have that $[\![A]\!]_{\hat{\pi}_k(\xi)}(x, \rho) = [\![\uparrow_k A]\!]_\xi (x, \rho')$ so we have that $a \in [\![\uparrow_k A]\!]_\xi (x, \rho')(id_x)$. It follows that $(\rho', a) \in \xi \cdot [\![\uparrow_k A]\!]_\xi$ and that $(\rho, a) \in \hat{\pi}_{(k+1)}\left(\xi \cdot [\![\uparrow_k A]\!]_\xi\right)$. We have one inclusion. The other inclusion is proven the same way, we only reverse the proof.

To complete the proof of this case we also have to prove that the morphisms of the two previous presheaves are equal, and that the equality part of the lemma holds. But this is immediate thanks to the induction hypothesis.

- $\uparrow_k \ \Pi\ A\ M = \Pi\ (\uparrow_k A)\ (\uparrow_{k+1} M)$

  Like in the previous case we have

$$[A\ type]_{\hat{\pi}_k(\xi)} \iff [\uparrow_k A\ type]_\xi$$

$$[M]_{\hat{\pi}_k(\xi)\cdot[\![A]\!]_{\hat{\pi}_k(\xi)}} \iff [\uparrow_{k+1} M]_{\xi\cdot[\![\uparrow_k A]\!]_\xi}$$

  and the associated equalities thanks to the induction hypothesis. The only difference between this case and the previous is that we have to show

$$[\![M]\!]_{\hat{\pi}_k(\xi)\cdot[\![A]\!]_{\hat{\pi}_k(\xi)}} \in type_{\hat{\pi}_k(\xi)\cdot[\![A]\!]_{\hat{\pi}_k(\xi)}} \iff [\![\uparrow_{k+1} M]\!]_{\xi\cdot[\![\uparrow_k A]\!]_\xi} \in type_{\xi\cdot[\![\uparrow_k A]\!]_\xi}$$

  before proving the goal equivalence and equality the same way as before. But this is straightforward rewriting using the induction hypothesis and the equalities that we can derive like in the previous case.

- $\uparrow_k \ \Sigma\ A\ M = \Sigma\ (\uparrow_k A)\ (\uparrow_{k+1} M)$

  We prove this case like the previous one.

- $\uparrow_k \ M, N = (\uparrow_k M), (\uparrow_k N)$

  Straightforward using the induction hypothesis.

- $\uparrow_k \ M.1 = (\uparrow_k M).1$

  Straightforward using the induction hypothesis. We have to use the equality part of the induction hypothesis to prove the goal equivalence, but because "no types are dependent" this is indeed straightforward.

- $\uparrow_k \ M.2 = (\uparrow_k M).2$

  We prove this case like the previous one.

- $\uparrow_k \ \mathcal{U}_l = \mathcal{U}_l$

  Immediate. Indeed, both sides are the same.

- $\uparrow_k \ \Omega = \Omega$

  Immediate.

- $\uparrow_k \ * = *$

  Immediate.

- $\uparrow_k \ M \equiv_A N = (\uparrow_k M) \equiv_{\uparrow_k A} (\uparrow_k N)$

  Straightforward using the induction hypothesis.

- $\|U\| [k \setminus N] = \|U [k \setminus N]\|$

  Straightforward using the induction hypothesis.

- $|U| [k \setminus N] = |U [k \setminus N]|$

  By induction if U is defined then $U [k \setminus N]$ is also defined. The result follows.

- $\langle T \rangle [k \setminus N] = \langle T [k \setminus N] \rangle$

  This is also straightforward.

A.2 Semantic substitution

Let be $n, k \in \mathbb{N}$, $S \in term_n$ and $\xi \in \widehat{\mathcal{C}}_{n+1+k}$ such that $[S]_{\pi_{->k}(\xi)}$ and

$$\forall x \in \mathcal{C},\ \forall \rho \in \xi_x,\ \pi_k(\rho) = [\![S]\!]_{\pi_{->k}(\xi)}(x, \pi_{->k}(\rho))$$

Then for all $M \in term_{n+1+k}$ we have

$$[M]_\xi \iff [M [k \setminus S]]_{\hat{\pi}_k(\xi)}$$
$$\forall x \in \mathcal{C},\ \forall \rho \in \xi_x,\ [\![M]\!]_\xi (x, \rho) = [\![M [k \setminus S]]\!]_{\hat{\pi}_k(\xi)} (x, \hat{\pi}_k(\rho))$$

We prove this by induction. Most of the cases (actually all but the variable cases) follows the same pattern as their corresponding case in the semantic weakening proof so we omit them here.

- $var_0 [0 \setminus S] = S$

  We first have to prove $[var_0]_\xi \iff [S]_{\hat{\pi}_0(\xi)}$. The left side is true by definition and the right one by hypothesis. The associated equality that we then have to prove is also true by hypothesis.

- $var_{i+1} [0 \setminus S] = var_i$

  Both side of the equivalence are true by definition. The equality follows by computation of both side.

- $var_0 [k+1 \setminus S] = var_0$

  Same as previous.

- $var_{i+1} [k+1 \setminus S] = \uparrow_0 (var_i [k \setminus S])$

  We first have to prove that

  $$[var_{i+1}]_\xi \iff [\uparrow_0 (var_i [k \setminus S])]_{\hat{\pi}_{(k+1)}(\xi)}$$

  Because the left side is true by definition we only have to prove that the right side is true. By the induction hypothesis we know that

  $$[var_i]_{\hat{\pi}_0(\xi)} \iff [var_i [k \setminus S]]_{\hat{\pi}_{\{0,k+1\}}(\xi)}$$

  which also implies that $[var_i [k \setminus S]]_{\hat{\pi}_{\{0,k+1\}}(\xi)}$ holds. We then use the weakening lemma to prove the equivalence of the right hand sides of both equations.

  The equality part of the lemma is also proved thanks to the weakening lemma.

A.3 Derivation rules

We prove here that the derivation rules from the theory in the first section are valid in any presheaf model.

Structural rules

$$\frac{}{\epsilon \vdash}$$

The empty context is always defined, and its semantic is the 0-product in $\widehat{\mathcal{C}}$.

$$\frac{\Gamma \vdash A\ type}{\Gamma, A \vdash}$$

The semantic predicate of the hypothesis and the conclusion are the same.

$$\frac{\Gamma, \Delta \vdash \mathcal{J} \qquad \Gamma \vdash A\ type}{\Gamma, A, \uparrow \Delta \vdash \uparrow_{|\Delta|} \mathcal{J}}$$

We prove by induction on Δ that the weakening context is well defined thanks to the weakening lemma. Then the same lemma then proves that the conclusion holds.

$$\frac{\Gamma, A, \Delta \vdash \mathcal{J} \qquad \Gamma \vdash M : A}{\Gamma, \Delta\,[M] \vdash \mathcal{J}\,[|\Delta| \setminus M]}$$

As in the previous case we first use the substitution lemma to prove by induction on Δ that the conclusion's context is defined. Then we conclude with the substitution lemma.

$$\frac{\Gamma, A \vdash}{\Gamma, A \vdash var_0 : A}$$

This is immediate.

Definitional equality

$$\frac{\Gamma \vdash A\ type}{\Gamma \vdash A = A\ type} \qquad \frac{\Gamma \vdash A = B\ type}{\Gamma \vdash B = A\ type} \qquad \frac{\Gamma \vdash A = B\ type \qquad \Gamma \vdash B = C\ type}{\Gamma \vdash A = C\ type}$$

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash M = N : A} \qquad \frac{\Gamma \vdash M = N : A}{\Gamma \vdash N = M\ A} \qquad \frac{\Gamma \vdash M = N : A \qquad \Gamma \vdash N = O : A}{\Gamma \vdash M = O : A}$$

The definitional equality on types and on terms is an equivalence relation because it is

$$\frac{\Gamma \vdash M = N : A \qquad \Gamma, \Delta\,[M] \vdash \mathcal{J}\,[M]}{\Gamma, \Delta\,[N] \vdash \mathcal{J}\,[N]}$$

We have that M and N are both defined and that there semantic is the same, so the conclusion holds.

$$\frac{\Gamma, A \vdash M = N : B \qquad \Gamma \vdash \lambda\ A\ M}{\Gamma \vdash \lambda\ A\ N}$$

$$\frac{\Gamma, A \vdash M = N : B \qquad \Gamma \vdash \Pi\ A\ M}{\Gamma \vdash \Pi\ A\ N}$$

$$\frac{\Gamma, A \vdash M = N : B \qquad \Gamma \vdash \Sigma\ A\ M}{\Gamma \vdash \Sigma\ A\ N}$$

These congruence rules are also immediate to prove thanks to the hypothesis that M and N have the same semantic.

Universes

$$\frac{\Gamma \vdash}{\Gamma \vdash \mathcal{U}_l\ type} \qquad \frac{\Gamma \vdash}{\Gamma \vdash \Omega\ type}$$

Universes are types by definition, so the conclusion reduces to the context being defined, which is the hypothesis.

$$\frac{\Gamma \vdash}{\Gamma \vdash \mathcal{U}_l : \mathcal{U}_{l+1}} \qquad \frac{\Gamma \vdash}{\Gamma \vdash \Omega : \mathcal{U}_0}$$

These rules holds thanks to the definition of Grothendieck universes. Indeed all the constructions made to construct a type theoretic universe $univ_n$ are made is $n+1$-th Grothendieck universe. The propositional universe is itself built in the first Grothendieck universe.

$$\frac{\Gamma \vdash A : \mathcal{U}_l}{\Gamma \vdash A\ type} \qquad \frac{\Gamma \vdash A : \Omega}{\Gamma \vdash A\ type}$$

To be a type is defined as to be an element of some universe but without the set theoretic size constraints, hence these rules hold.

$$\frac{\Gamma \vdash A : \mathcal{U}_l}{\Gamma \vdash A : \mathcal{U}_{l+1}} \qquad \frac{\Gamma \vdash A : \Omega}{\Gamma \vdash A : \mathcal{U}_0}$$

The only differences being the universes are size constraints. Those rules weaken these constraints.

Product types

$$\frac{\Gamma, A \vdash B\ type}{\Gamma \vdash \Pi\ A\ B\ type}$$

It is immediate to see that $[\Pi\ A\ B]_{[\![\Gamma]\!]}$, and it is also immediate to see from its definition that, $\forall x, y \in \mathcal{C}$, $\forall \rho \in [\![\Gamma]\!]_x$, $f \in \mathcal{C}(y, x)$, we have $[\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]}(x, \rho) \in Cat(\mathcal{C}/x, Set)$.

Let be $x, y, z \in \mathcal{C}$, $f \in \mathcal{C}(y, x)$, $g \in \mathcal{C}(z, y)$ and $\rho \in [\![\Gamma]\!]_x$.

$$\begin{aligned}
&[\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]}(y, \rho \cdot f)(g)\\
&= (h \in \Sigma w \mathcal{C}(w, z)) \to \Big(a \in [\![A]\!]_{[\![\Gamma]\!]}(w, \rho \cdot f \cdot g \cdot h)(id_w)\Big) \to\\
&\quad [\![M]\!]_{[\![\Gamma]\!] \cdot [\![A]\!]_{[\![\Gamma]\!]}}(w, (\rho \cdot f \cdot g \cdot h, a))(id_w)\\
&= (h \in \Sigma w \mathcal{C}(w, z)) \to \Big(a \in [\![A]\!]_{[\![\Gamma]\!]}(w, \rho \cdot (f \circ g) \cdot h)(id_w)\Big) \to\\
&\quad [\![M]\!]_{[\![\Gamma]\!] \cdot [\![A]\!]_{[\![\Gamma]\!]}}(w, (\rho \cdot (f \circ g) \cdot h, a))(id_w)\\
&= [\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]}(x, \rho)(f \circ g)
\end{aligned}$$

$$\begin{aligned}
&[\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]}(y, \rho \cdot f)\left(\overleftarrow{g}^{id_y}\right)\\
&= \Big(e \in [\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]}(y, \rho \cdot f)(id_y)\Big) \mapsto (h \in \Sigma w \mathcal{C}(w, z)) \mapsto e(g \circ h)\\
&= \Big(e \in [\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]}(x, \rho)(f)\Big) \mapsto (h \in \Sigma w \mathcal{C}(w, z)) \mapsto e(g \circ h)\\
&= [\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]}(x, \rho)\left(\overleftarrow{g}^{f}\right)
\end{aligned}$$

With this we have $[\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]} \in type_{[\![\Gamma]\!]}$, and thus the validity of the rule.

$$\frac{\Gamma \vdash A : \mathcal{U}_l \qquad \Gamma, A \vdash B : \mathcal{U}_l}{\Gamma \vdash \Pi\ A\ B : \mathcal{U}_l}$$

We have all the conclusion but the size constraints from the previous derivation rule. These constraints are satisfied because only constructions under which universes are stable are used, so that the result is still in the l-th Grothendieck universe.

$$\frac{\Gamma, A \vdash M : B}{\Gamma \vdash \lambda\ A\ M : \Pi\ A\ B}$$

Likewise almost everything is immediate. We only need one more equality to prove that $[\![\lambda\ A\ M]\!] \in type_{[\![\Gamma]\!], [\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]}}$.

$$\begin{aligned}
&[\![\Pi\ A\ B]\!]_{[\![\Gamma]\!]}(x, \rho)(f : id_x \to f)\Big([\![\lambda\ A\ M]\!]_{[\![\Gamma]\!]}(x, \rho)\Big)\\
&= (g \in \Sigma z \mathcal{C}(z, y)) \mapsto [\![\lambda\ A\ B]\!]_{[\![\Gamma]\!]}(x, \rho)(f \circ g)\\
&= (g \in \Sigma z \mathcal{C}(z, y)) \mapsto [\![\lambda\ A\ B]\!]_{[\![\Gamma]\!]}(y, \rho \cdot f)(g)\\
&= [\![\lambda\ A\ B]\!]_{[\![\Gamma]\!]}(y, \rho \cdot f)
\end{aligned}$$

$$\frac{\Gamma \vdash M : \Pi\ A\ B \qquad \Gamma, A \vdash N : B}{\Gamma \vdash M\ N : B[N]}$$

The fact that $M\ N$ is well defined is immediate from the hypothesis. Also, $\Gamma \vdash B[N]$ is defined thanks to the substitution lemma. We then recover from the semantic of a Π-type that the codomain of the application that defines $M\ N$ is $B[N]$.

$$\frac{\Gamma, A \vdash M : B \qquad \Gamma \vdash N : A}{\Gamma \vdash (\lambda\ A\ M)\ \ N = M[N] : B[N]}$$

Substituted terms are defined thanks to the substitution lemma. The rule then follows from computations of the semantics.

Sum types

$$\frac{\Gamma, A \vdash B\ type}{\Gamma \vdash \Sigma\ A\ B\ type} \quad \frac{\Gamma \vdash A : \mathcal{U}_l \qquad \Gamma, A \vdash B : \mathcal{U}_l}{\Gamma \vdash \Sigma\ A\ B\ : \mathcal{U}_l}$$

We show the first of the two derivation rule, the other will follow easily by looking at the size of the semantics.

It is immediate to see that $[\Sigma\ A\ B]_{[\![\Gamma]\!]}$ holds. Let be $\xi := [\![\Gamma]\!]$. Let $x, y, z \in \mathcal{C}$, $\rho \in \xi_x$, $f \in \mathcal{C}(y, x)$, $g \in \mathcal{C}(z, y)$ and $(a, b) \in [\![\Sigma\ A\ B]\!]_\xi (x, \rho)(f)$. We have

$$\begin{aligned} & [\![A]\!]_\xi (y, \rho \cdot f)\left(\overleftarrow{g}^{id_y}\right)(a) \\ & \quad \in \ [\![A]\!]_\xi (y, \rho \cdot f)(g) \\ & \quad = \ [\![A]\!]_\xi (z, \rho \cdot f \cdot g)(id_z) \end{aligned}$$

$$\begin{aligned} & [\![B]\!]_{\xi \cdot [\![A]\!]_\xi} (y, (\rho \cdot f, a))\left(\overleftarrow{g}^{id_y}\right)(b) \\ & \quad \in \ [\![B]\!]_{\xi \cdot [\![A]\!]_\xi} (y, (\rho \cdot f, a))(g) \\ & \quad = \ [\![B]\!]_{\xi \cdot [\![A]\!]_\xi} \left(z, \left(\rho \cdot f \cdot g, [\![A]\!]_\xi (y, \rho \cdot f)\left(\overleftarrow{g}^{id_y}\right)(a)\right)\right)(id_z) \end{aligned}$$

Hence $[\![\Sigma\ A\ B]\!]_\xi (x, \rho)\left(\overleftarrow{g}^{f}\right)(a, b) \in [\![\Sigma\ A\ B]\!]_\xi (x, \rho)(f \circ g)$ and $[\![\Sigma\ A\ B]\!]_\xi (x, \rho) \in Cat(\mathcal{C}/x, V)$. The two other results needed to have $[\![\Sigma\ A\ B]\!]_\xi \in type_\xi$ are immediate thanks to the definitions.

$$\frac{\Gamma, A \vdash B\ type \qquad \Gamma \vdash M : A \qquad \Gamma \vdash N : B[M]}{\Gamma \vdash M, N : \Sigma\ A\ B}$$

It is immediate to see that $[M, N]_\xi$ and $[\![\Sigma\ A\ B]\!]_\xi \in type_\xi$, where $\xi := [\![\Gamma]\!]$. Let $x \in \mathcal{C}$, $\rho \in \xi_x$. Because of the substitution lemma we have that

$$[\![B]\!]_{\xi \cdot [\![A]\!]_\xi} \left(x, \left(\rho, [\![M]\!]_\xi (x, \rho)\right)\right)(id_x) = [\![B[M]]\!]_\xi (x, \rho)(id_x)$$

It follows that $[\![M, N]\!]_\xi (x, \rho) \in [\![\Sigma\ A\ B]\!]_\xi (x, \rho)(id_x)$. It is then straightforward to see that, $\forall x, y \in \mathcal{C}$, $\forall f \in \mathcal{C}(y, x)$, $\forall \rho \in \xi_x$,

$$[\![\Sigma\ A\ B]\!]_\xi (x, \rho)\left(\overleftarrow{f}^{id_x}\right)\left([\![M, N]\!]_\xi (x, \rho)\right) = [\![M, N]\!]_\xi (u, \rho \cdot f)$$

$$\frac{\Gamma \vdash M : \Sigma\ A\ B}{\Gamma \vdash M.1 : A} \qquad \frac{\Gamma, A \vdash B\ type \qquad \Gamma \vdash M : A \qquad \Gamma \vdash N : B\,[M]}{\Gamma \vdash (M, N)\,.1 = M : A}$$

$$\frac{\Gamma \vdash M : \Sigma\ A\ B}{\Gamma \vdash M.2 : B\,[M.1]} \qquad \frac{\Gamma, A \vdash B\ type \qquad \Gamma \vdash M : A \qquad \Gamma \vdash N : B\,[M]}{\Gamma \vdash (M, N)\,.2 = N : B\,[M]}$$

These rules are straightforward to prove. We still need the substitution lemma to have that the substituted terms are well-defined.

Propositions

$$\frac{\Gamma \vdash A : prop \qquad \Gamma \vdash M : A}{\Gamma \vdash M = * : A}$$

$$\frac{\Gamma \vdash A : \Omega \qquad \Gamma \vdash B : \Omega \qquad \Gamma, x : A \vdash b : B \qquad \Gamma, x : B \vdash a : A}{\Gamma \vdash A = B : \Omega}$$

By unfolding the last two hypothesis we have that at each stage whenever A or B is inhabited then the other is. It follows that their squashed types are equal, but A and B being in *prop* they are equal to their squashing.

Squash types

$$\frac{\Gamma \vdash A\ type}{\Gamma \vdash \|A\| : \Omega}$$

Notice that, $\forall x, y, z \in \mathcal{C}$, $\forall \rho \in [\![\Gamma]\!]_x$, $\forall f \in \mathcal{C}\,(y, x)$, $\forall g \in \mathcal{C}\,(z, y)$,

$$[\![A]\!]_{[\![\Gamma]\!]}\,(x, \rho)\left(\overleftarrow{g}^{\,f}\right) : [\![A]\!]_{[\![\Gamma]\!]}\,(x, \rho)\,(f) \to [\![A]\!]_{[\![\Gamma]\!]}\,(x, \rho)\,(f \circ g)$$

hence

$$[\![A]\!]_{[\![\Gamma]\!]}\,(x, \rho)\,(f) \neq 0 \implies [\![A]\!]_{[\![\Gamma]\!]}\,(x, \rho)\,(f \circ g) \neq 0$$

The validity of the rule is then straightforward to prove.

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash |M| : \|A\|}$$

The rule follows from the definition of a squashed type.

$$\frac{\Gamma \vdash N : A \to B \qquad \Gamma, A, A \vdash N\ var_1 = N\ var_0 : B}{\Gamma \vdash \langle N \rangle : \|A\| \to B}$$

N being a constant function we have that $\langle N \rangle$ is defined. We get its domain and thus its type from N.

$$\frac{\Gamma \vdash N : A \to B \qquad \Gamma, A, A \vdash N\ var_1 = N\ var_0 : B \qquad \Gamma \vdash M : A}{\Gamma \vdash \langle N \rangle\ \ |M| = N\ M : B}$$

This is immediate by looking at the semantics, knowing that N is a constant function.

Propositional equality

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash N : A}{\Gamma \vdash M \equiv_A N : \Omega} \qquad \frac{\Gamma \vdash M = N : A}{\Gamma \vdash * : M \equiv_A N} \qquad \frac{\Gamma \vdash T : M \equiv_A N}{\Gamma \vdash M = N : A}$$

These rules are straighforward to prove from the definitions.

B Proofs from section 2

B.1 Extension structure in dependent type theory

We suppose a type $\mathbb{T}$ of propositions and a strictness axiom for each universe level, with $\cong$ the type of isomorphisms, like the one stated in [OP17] for cofibrant propositions

$$\begin{aligned} \vdash_1 \texttt{strict}_l : \quad & \Pi\,(\varphi : \mathbb{T})\,\Pi\,(A : \varphi \to \mathcal{U}_l)\,\Pi\,(B : \mathcal{U}_l)\,\Pi\,(s : \Pi\,\varphi\,(A * \cong B)) \\ & \Sigma\,(B' : \mathcal{U}_l)\,\Sigma\,(s' : B' \cong B)\,\Pi\,\varphi\,(A * = B' \wedge s * = s') \end{aligned}$$

Our aim will be to translate into a dependent type theory $\vdash_1$ a theory $\vdash_2$ in which partial elements on the extent of a proposition in $\mathbb{T}$ can be extended. Types $\vdash_2$ will be the types A in $\vdash_1$ equipped with an extension structure $\texttt{Ext}_A$ that will be translated into

$$\Pi\,(\varphi : \mathbb{T})\,\Sigma\,(\rho : \Pi\ \varphi\ A \to A)\,(\varphi \to \Pi\,(a : \Pi\ \varphi\ A)\ \ \rho\ a = a\ *)$$

In order to prove such translation we will only have to prove that the extension structure is preserved by the type formers in $\vdash_1$. Hence we will work only in $\vdash_1$.

Product types Let A be a type that may not have extension structures and B a type on A such that, for all $a : A$, $B\ a$ has an extension structure $\rho_B\ a$. Assume $f : \varphi \to \Pi\ A\ B$ with $\varphi : \mathbb{T}$. $\lambda\,(a : A)\ \ \rho_B\ a\ \ (\lambda\ \varphi\ (f\ *\ a))$ is an extension of f.

Sum types Let A be a type equipped with an extension structure ρ_A and B a type of A also equipped with an extension structure ρ_B. Let $\varphi : \mathbb{T}$ and $t : \varphi \to \Sigma\ A\ B$. First we extend the first projection of t, this defines $t_1' := \rho_A\ \ (\lambda\ \varphi\ \ (t\ *\ .1)) : A$. Because t_1' and $t\ *\ .1$ are equal on φ then $B\ t_1'$ and $B\ \ (t\ *\ .1)$ are also the same on φ. Hence we can use $\rho_B\ t_1'$ to extend the second projection of t. This defines an extension for t.

Predicative universes Let $l \in \mathbb{N}$ be a universe level. We build an extension structure for $univ_l$. Assume $\varphi : \mathbb{T}$ and $A : \varphi \to \mathcal{U}_l$. We define $B := \Pi\ \varphi\ A$ and $T := \texttt{strict}_l\ \varphi\ A\ B\ s : \Sigma\,(B' : \mathcal{U}_l)\,\Sigma\,(s' : B' \cong B)\,\Pi\ \varphi\ (A* = B' \wedge s* = s')$ with s being the trivial isomorphisms between $A*$ and B over φ. The first projection of T defines an extension for A.

Proposition universe Whatever the propositions $\varphi : \mathbb{T}$ and $\psi : \varphi \to \Omega$ we have that $\|\Pi\ \varphi\ \psi\|$ is an extension of ψ.

B.2 Extension structure in cubical type theory

We use the notation $\langle \varphi_1 \mapsto a_1 \; ; \; \dots \; ; \; \varphi_n \mapsto a_n \rangle$ to denote the system defined on $\varphi_1 \vee \dots \vee \varphi_n$ whose value at φ_i is a_i for each i in $\{1, \dots, n\}$.

Stability by isomorphism An helpful lemma will be the stability of extension structures by strict isomorphism. Let A be a type with an extension ρ and a type B strictly isomorphic to A. Let $f : A \to B$ and $g : B \to A$ be the two side of the isomorphism. We use the isomorphism to lift the extension operation (without the equality constraint) from A to B. This defines a term

$$p := \lambda\,(\varphi : \mathbb{T})\;\; \lambda\,(a : \Pi\;\varphi\; A)\, f\,(\rho\;\varphi\,(\lambda\;\varphi\;g\;\;(a\;*))) : \Pi\,(\varphi : \mathbb{T})\,\Pi\;\varphi\;A \to A$$

and we have on the extent of $\varphi : \mathbb{T}$ that

$$\begin{aligned} p\;\varphi\;a &= f\,(\rho\;\varphi\,(\lambda\;\varphi\;g\;\;(a\;*))) \\ &= f\,(g\,(a\;*)) \\ &= a\;* \end{aligned}$$

Hence p defines an extension structure for B.

Path types We define the type $Path\;A\;a\;b := \Sigma\,(p : \mathbb{I} \to A)\;\; p\;\mathtt{0} = a \wedge p\;\mathtt{1} = b$. Assume that A as an extension structure ρ and terms $\varphi : \mathbb{T}$ and $p : \varphi \to Path\;A\;a\;b$ with a and b defined globally. By hypothesis on $\mathbb{T}$, if we add an element $i : \mathbb{I}$ to the context then the proposition $\psi := \varphi \vee i = \mathtt{0} \vee i = \mathtt{1}$ is still in $\mathbb{T}$. We define

$$s := \lambda\,(i : \mathbb{I})\;\; \rho\;\psi\;\; (\lambda\;\psi\;\; \langle \varphi \mapsto (p\;*\;i)\;;\; i = \mathtt{0} \mapsto a\,;\; i = \mathtt{1} \mapsto b \rangle) : \mathbb{I} \to A$$

We have by definition of ρ that $s\;\mathtt{0} = a$ and $s\;\mathtt{1} = b$ so that s defines an element of $Path\;A\;a\;b$. Moreover we have that this path is equal to p on φ.

Glue types Because of the stability of extension structures by isomorphism we only need to be able to lift the type

$$G = Glue\;\varphi\;A\;B\;f := \Sigma\,(a : \varphi \to A\;*)\,\Sigma\,(b : B)\,\varphi \to f\;\;*\;\;(a\;*) = b$$

with $\varphi : \mathbb{F}$, $A : \varphi \to \mathcal{U}$, $B : \mathcal{U}$ and $f : (u : \varphi) \to A\;u \to B$, to be able to lift the strict glue type from [OP17].

Assume that A and B have extension structures ρ_A and ρ_B and that we have terms $\psi : \mathbb{T}$ and $g : \psi \to G$. By postcomposing g with the right projections we get the three following terms

$$\begin{aligned} &a : \psi \to \varphi \to A\;* \\ &b : \psi \to B \\ &e : \psi \to \varphi \to f\;*\;(a\;*\;*) = b\;* \end{aligned}$$

We first extend a into

$$a' := \lambda\,(\_ : \varphi)\ \rho_A\ *\ \psi\ \ (\lambda\,(\_ : \psi)\ \ a\ *\ *) : \varphi \to A\ *$$

We then extend b on $\psi\ \vee\ \varphi : \mathbb{T}$ in order to preserve the equality constraint. Thus we get

$$b' := \rho_B\ \ (\psi \vee \varphi)\,\langle \psi \mapsto b\ *\ ;\ \varphi \mapsto f\ *\ (a'\,*)\rangle : B$$

On ψ this restricts to b and on φ it implies the equality constraint of glue types.

Composition structures It remains to prove that we can extend composition structures for type families as in [OP17]. Let B be a type family on a type A. A composition structure for B is a function

$$\begin{aligned} c_B :\quad & \Pi\,(\varphi : \mathbb{F})\,\Pi\,(f : \mathbb{I} \to A)\,\Pi\,(p : \varphi \to \Pi\,(i : \mathbb{I})\,B\,(p\ i)) \\ & \Pi\,(x_0 : B\ \ (p\ \mathtt{0}))\,(\varphi \to p\ *\ \mathtt{0} = x_0) \to \Sigma\,(x_1 : B\ \ (p\ \mathtt{1}))\,\varphi \to p\ *\ \mathtt{1} = x_1 \end{aligned}$$

Consider that we have a composition structure for B on the extent of $\psi : \mathbb{T}$ and that we have an extension structure for the type family B. We use here the same trick as before, that is to say that we construct the result of the composition structure we extend the system on $\psi \vee \varphi$ made by x_1 on ψ and $p\ *\ \mathtt{1}$ on φ (by hypothesis x_1 may not be defined on φ).