A predicative variant of a realizability tripos for the Minimalist Foundation

Maria Emilia Maietti
Dipartimento di Matematica, Università di Padova
maietti@math.unipd.it

Samuele Maschio
Dipartimento di Matematica, Università di Padova
maschio@math.unipd.it

Abstract

Here we present a predicative variant of a realizability tripos validating the intensional level of the Minimalist Foundation extended with Formal Church thesis, for short **CT**.

The original concept of tripos was introduced in the 80s by J. M. E. Hyland, P. T. Johnstone and A. M. Pitts in order to build various kinds of toposes including realizability ones.

Our categorical structure provides the key ingredient to build a predicative variant of a realizability topos satisfying **CT**, like Hyland's Effective topos, where to validate the extensional level of the Minimalist Foundation.

The adjective *predicative* refers to the fact that our categorical structure is formalized in Feferman's theory of inductive definitions $\widehat{ID_1}$.

1 Introduction

Constructive mathematics is mathematics developed with constructive proofs, that is proofs enjoying a computational method to construct witnesses of their existential statements. As a consequence constructively definable number theoretic functions

Thank the referees (or not), and anyone else for reasons relating to the paper as a whole, by adding \titlethanks{text} *to the preamble* (or \titlethanks{} to delete this reminder).

For acknowledgements specific to the author (e.g. thanking a particular author's funding body), use \thanks{text} after the author name in the \addauthor command (or do not add the command at all for no such addition).

are all computable. It is indeed often said that constructive mathematics is abstract mathematics which is implicitely computable.

To give evidence of such a claim it is convenient to call “constructive” those proofs that can be formalized in a foundation enjoying a so called “realizability model” where one may extract the computational contents of its proofs by interpreting its sets as data types and its functions as programs. The most basic example of such a constructive foundation, at least for constructive arithmetics, is Intuitionistic Arithmetic HA in [40]. Its realizability semantics is the well-known Kleene realizability interpretation (see for example [40]) which makes HA consistent with the so called Formal Church thesis, for short **CT**, expressing that from a total number-theoretic relation we can extract a computable function. Actually, most constructive foundations in the literature are consistent with **CT** and this is the case also for the Minimalist Foundation.

The Minimalist Foundation, for short **MF**, was conceived by the first author in joint work with G. Sambin in [28] as a common core among the most relevant constructive and classical foundations, introduced both in type theory, in category theory and in axiomatic set theory. In [28] **MF** is also required to be a two-level system equipped with an *intensional level* suitable for extraction of computational contents from its proofs, an *extensional level* formulated in a language as close as possible to that of ordinary mathematics and an interpretation of the latter in the former showing that the extensional level has been obtained by abstraction from the intensional one according to Sambin's forget-restore principle in [35].

A two-level formal system of this kind for **MF** was completed in [23]. Both intensional and extensional levels of **MF** consist of type systems based on versions of Martin-Löf's type theory with the addition of a primitive notion of propositions and some related constructors: the intensional one, called **mTT**, is based on [31] and the extensional one, called **emTT**, on [30]. Actually, **mTT** can be considered a *predicative* version of Coquand's Calculus of Constructions [13].

The two-level structure of **MF** has various kinds of benefits.

First of all it provides a framework for computer-aided formalization of its constructive proofs. Indeed the intensional level of **MF** has enough decidable properties to be a base for a proof-assistant in which to formalize the constructive proofs done at the extensional level via the interpretation provided in [23].

Moreover, the presence of two levels is crucial to easily show the compatibility of **MF** with the other foundations at the “right” level: the intensional level of **MF** can be easily interpreted in intensional theories such as those formulated in type theory, for example Martin-Löf's type theory [31] or Coquand's Calculus of Constructions, while its extensional level can be easily interpreted in extensional theories such as those formulated in axiomatic set theory, for example Aczel's constructive set theory [4], or those formulated in category theory, for example the internal languages of topoi or pretopoi [21, 22].

Finally, the two levels of **MF** and their link resemble a well-known construction in category theory, namely the tripos-to-topos construction of a realizability topos in [18]. This is because the interpretation of the extensional level of **MF** in [23] is done in a *quotient completion* built on the intensional level of **MF**. Such a quotient completion had been studied categorically in [26], [25] under the name of "elementary quotient completion" and related to the well known notion of exact completion on a lex or regular category in [27]. Then, an analogy between **MF** and the tripos-to-topos construction of a realizability topos can be described as follows: the categorical structure of the intensional level of **MF** plays the role of a tripos, its elementary quotient completion plays the role of the realizability topos construction, while the extensional level of **MF** plays the role of the internal language of a generic elementary topos.

In this paper we strengthen this analogy by building a realizability categorical structure for the intensional level **mTT** of **MF** in Feferman's classical predicative theory of inductive definitions $\widehat{ID_1}$ (see e. g. [15]). This is obtained by extracting the categorical structure behind the realizability interpretation in [24] for **mTT** in $\widehat{ID_1}$. As an advantage we get an easier proof of validity for **mTT** by defining a partial typed interpretation as in [38].

Our categorical semantics for **mTT** is called "effective" since it validates the formal Church thesis and constitutes the key ingredient to build a predicative variant of Hyland's "Effective Topos" [17] in $\widehat{ID_1}$, where to interpret the extensional level of **MF** extended with **CT**.

A predicative study of the Effective Topos, and more generally of realizability toposes, had been already developed in the context of algebraic set theory by B. van den Berg and I. Moerdijk, in particular in [41], by taking Aczel's Constructive Zermelo-Fraenkel set theory (for short CZF) in [4] as the predicative constructive set theory to be realized in their categorical structure.

A precise comparison between our work and that in [41] is expected to mirror the relationship between **MF** and CZF described in [23] and it is left to future work. We just recall that **mTT**, and the whole **MF**, is a much weaker theory than CZF concerning the proof-theoretic strength, because it can be interpreted in a strictly predicative theory as Feferman's $\widehat{ID_1}$ as [24] shows, while CZF is known not to be a predicative theory in Feferman's sense (see [16], [1], [11]).

2 The Minimalist Foundation

A peculiarity of constructive mathematics with respect to classical mathematics is the absence of a commonly accepted standard foundation as Zermelo-Fraenkel set theory for classical mathematics.

Various logical systems are available in the literature to formalize constructive mathematics: they range from axiomatic set theories à la Zermelo-Fraenkel, such as Aczel's CZF [4, 1, 2, 3] or Friedman's IZF [8], to the internal set theory of categorical universes such as topoi or pretopoi [21, 19, 22], to type theories such as Martin-Löf's type theory [31] or Coquand's Calculus of Inductive Constructions [13, 14]. No existing constructive foundation has yet superseded the others as the standard one.

The Minimalist Foundation, for short **MF**, was conceived in [28] to serve as a common core among the most relevant constructive and classical foundations. A key novelty of **MF** required in [28] is to be a two-level formal system equipped with an *intensional level* suitable for extraction of computational contents from its proofs, an *extensional level* formulated in a language as close as possible to that of ordinary mathematics and an interpretation of the latter in the former showing that the extensional level is obtained by abstraction from the intensional one according to Sambin's forget-restore principle in [35].

The two-level formal system of **MF** was completed in [23] with an interpretation of the extensional level into a quotient model of the intensional level analyzed categorically in [26, 25, 27].

The two-level structure of **MF** has at least two main advantages. On one hand the compatibility of **MF** with the most relevant constructive and classical foundations can be done at the most suitable level, namely the intensional level with intensional foundations, mostly designed as type theories, and the extensional one with usual extensional foundations, mostly designed as axiomatic set theories. On the other hand the two-level structure of **MF** has the advantage to meet the usual practice of developing mathematics in an extensional set theory, represented by the extensional level of **MF**, whose equalities are undecidable, with the practice of formalizing mathematical proofs in a computer-aided way by means of an interactive proof-assistant based on intensional type theory with decidable properties, including decidable type-checking of proofs and equalities, such as for example Agda [10] (on Martin-Löf's type theory [31]), or Coq [12, 9] or Matita [6, 5] (on the Calculus of Inductive Constructions).

2.1 Distinct features of MF

Here we present the main distinct conceptual features of both levels of **MF**. Their design is certainly influenced by the need of building a foundation for constructive mathematics which is compatible with the most relevant constructive and classical foundations at the appropriate level. An immediate consequence is that both levels of **MF** must be *predicative* theories to be compatible with well-known predicative theories, such as Martin-Löf's type theory or Aczel's CZF.

To meet this goal one could think of using Heyting arithmetic, possibly extended with finite types, as the extensional level for **MF**. However, in order to formalize most of constructive mathematics, and in particular constructive topology, in an extensional language close to that used in common practice, it would be good to have a theory with a more expressive language including quotient sets and the power of any set. On the other hand, it is worth noting that the power of a non-empty set inherits an impredicative nature as soon as it is considered a set and hence in a predicative set theory it must be considered an entity greater than a set, like a collection or a class. This fact led to introduce the notion of *collection* beside that of *set* at both levels of **MF**.

Concerning the intensional level of **MF**, the authors in [28] thought of designing it as an intensional dependent type theory à la Martin-Löf like that in [31]. Then, to make the extensional level interpretable in the intensional one easily and in a modular way, in [23] also the extensional level was designed as a dependent theory à la Martin-Löf like that in [30].

The final outcome in [23] was to design the intensional level of **MF**, called **mTT**, as a **predicative version** of Coquand's Calculus of Constructions in [13], for short CoC, which is essentially the basic system behind the proof-assistants Coq [12, 9] and Matita [6, 5].

The main features of CoC and of its extension in Coq that are strictly connected with the design of **mTT** are the following:

- sets include *sets in first-order intensional Martin-Löf's type theory* (i.e. the fragment of Martin-Löf's type theory in [31] corresponding to first-order logic with list types but without universes or well-founded sets) and there is a *primitive notion of propositions*, closed under intuitionistic connectives and quantifiers and equipped with proof-terms; hence propositions are though of as sets of their proofs;

- there is a *universe of propositions*, which is a set in CoC.

It is worth noting that only the second feature makes CoC impredicative. This feature allows one to represent the power of a set in a suitable model of quotients, called the setoid model (see [7]).

It is also important to recall that the CoC-universe of propositions is inconsistent with an identification of sets as propositions typical of Martin-Löf's type theory (see [13]), for short MLtt. As a consequence, the existential quantifier of CoC can not be that in MLtt and it does not yield choice principles, like the axiom of choice (see [31, 30]), as shown in [39].

In particular, a relevant consequence of the above features of CoC, which is a peculiar feature of **MF** discussed in [29, 34], is the possibility of distinguishing between the notion of a type theoretic function between sets A, B

$$f \in A \to B$$

called operation in **MF** (see [29]), and the notion of functional relation determined by a relation $R(x, y)\, prop\, [\, x \in A, y \in B\,]$ for which we can prove

$$\forall x \in A\ \exists!\, y \in B\ R(x, y)$$

Indeed in CoC, as well as in **MF**, the so called *axiom of unique choice*

$$(\ \mathbf{AC!}\ ) \qquad \forall x \in A\ \exists!\, y \in B\ R(x, y)\ \longrightarrow\ \exists f \in A \to B\ \forall x \in A\ R(x, f(x))$$

which allows one to extract a type theoretic function from a functional relation, is not valid (see [39] for a proof). This distinction between type-theoretic functions and functional relations, beside the non-validity of **AC!**, is also a property of Feferman's theories in [15].

The design of **mTT** in [23] proposes a way to turn the mentioned features of CoC in a predicative form by *extending first-order Martin-Löf's intensional type theory in [31]* with

- a notion of *collection* beside that of *set*: collections include sets but also certain types that can not be considered sets predicatively;

- a primitive notion of *proposition* closed under intuitionistic connectives and quantifiers over both sets and collections;

- a notion of *small proposition* denoting propositions closed under intuitionistic connectives and quantifiers restricted to sets;

- proof-terms for all propositions: *small propositions* are defined as *sets* of their proofs, while generic *propositions* are defined as *collections* of their proofs;

- a collection of small propositions and a collection of propositional functions on any set;

The last feature is what in the quotient model in [23] allows to define a power-collection of a set A as a suitable quotient on the collection of propositional functions on A.

Accordingly, the extensional level of **MF** in [23], called **emTT**, is an extension of the extensional version of first-order Martin-Löf's type theory in [30] with the following distinct features:

- a notion of *collection* beside that of *set* as in the intensional level of **MF**;
- a primitive notion of *proposition*, closed under intuitionistic connectives and quantifiers over collections and a notion of *small proposition* denoting propositions closed under intuitionistic connectives and quantifiers restricted to sets;
- *proof-irrelevance* of all propositions, namely all propositions are equipped with at most a canonical proof-term to denote when they are true;
- *a power-collection for each set* where subsets are equivalence classes of small propositions depending on the set and quotiented under equiprovability;
- *effective quotient sets* of equivalence relations defined by small propositions.

An important consequence of **MF**-design is *the compatibility of* **MF** *with classical predicative theories as Feferman's predicative theories [15].* Indeed it is well known that the *addition of the principle of excluded middle* can turn a predicative theory where functional relations between sets form a set, as Aczel's CZF or Martin-Löf's type theory, into an impredicative one where *power-collections become sets.*

In the next we are going to describe in more details the type theory **mTT** of the intensional level and we refer to [23] for the description of the type theory **emTT** of the extensional level and of its interpretation in **mTT**.

2.2 The intensional level of the Minimalist Foundation

Here we describe the type theory **mTT** representing the intensional level of **MF** in [23], which extends that presented in [28]. **mTT** is a dependent type theory written in the style of Martin-Löf's type theory [31] by means of the following four kinds of judgements:

$$A\ type\ [\Gamma] \quad A = B\ type\ [\Gamma] \quad a \in A\ [\Gamma] \quad a = b \in A\ [\Gamma]$$

that is the type judgement (expressing that something is a specific type), the type equality judgement (expressing that two types are equal), the term judgement (expressing that something is a term of a certain type) and the term equality judgement (expressing the *definitional equality* between terms of the same type), respectively, all under a context Γ.

The word *type* is used as a meta-variable to indicate four kinds of entities: collections, sets, propositions and small propositions, namely

$$type \in \{col, set, prop, prop_s\}$$

Therefore, in **mTT** types are actually formed by using the following judgements:

$$A\ set\ [\Gamma] \qquad B\ col\ [\Gamma] \qquad \phi\ prop\ [\Gamma] \qquad \psi\ prop_s\ [\Gamma]$$

saying that A is a set, that B is a collection, that ϕ is a proposition and that ψ is a small proposition.

Here as in [24], and contrary to [23] where we use only capital latin letters as meta-variables for types, we use greek letters ψ, ϕ as meta-variables for propositions and capital latin letters A, B as meta-variables for sets or collections.

As in the intensional version of Martin-Löf's type theory, in **mTT** there are two kinds of equality concerning terms: one is the definitional equality of terms of the same type given by the judgement

$$a = b \in A\ [\Gamma]$$

which is decidable, and the other is the propositional equality written

$$\mathsf{Id}(A, a, b)\ prop\ [\Gamma]$$

which is not necessarily decidable.

We now proceed by briefly describing the various kinds of types in **mTT**, starting from small propositions and propositions and then passing to sets and finally collections.

Small propositions in **mTT** include all the logical constructors of intuitionistic predicate logic with equality and quantifications restricted to sets:

$$\phi\ prop_s \equiv \bot \mid \phi \wedge \psi \mid \phi \vee \psi \mid \phi \rightarrow \psi \mid (\forall x \in A)\, \phi \mid (\exists x \in A)\, \phi \mid \mathsf{Id}(A, a, b)$$

provided that A is a set.

Then, *propositions* in **mTT** include all the logical constructors of intuitionistic predicate logic with equality and quantifications on all kinds of types, i. e. sets and collections. Of course, small propositions are also propositions. Propositions can be generated as follows:

$$\phi\ prop\ \equiv \phi\ prop_s \mid \phi \wedge \psi \mid \phi \vee \psi \mid \phi \rightarrow \psi \mid (\forall x \in D)\, \phi \mid (\exists x \in D)\, \phi \mid \mathsf{Id}(D, d, b)$$

In order to close sets under comprehension, for example to include the set of positive natural numbers $\{x \in \mathsf{N} \mid x \geq 1\}$, and to define operations on such sets, we need to think of propositions as types of their proofs: small propositions are seen as sets of their proofs while generic propositions are seen as collections of their proofs. That is, we add to **mTT** the following rules

$$\textbf{prop}_s\textbf{-into-set)}\quad \frac{\phi\ prop_s}{\phi\ set} \qquad\qquad \textbf{prop-into-col)}\quad \frac{\phi\ prop}{\phi\ col}$$

Before explaining the differences between the notion of set and that of collection we describe their constructors in **mTT**.

Sets in **mTT** are characterized as inductively generated types and they include the following:

$$A\ set\ \equiv \phi\ prop_s \mid \mathsf{N}_0 \mid \mathsf{N}_1 \mid \mathsf{N} \mid \mathsf{List}(A) \mid (\Sigma x \in A)\, B \mid A + B \mid (\Pi x \in A)\, B$$

where the notation N_0 stands for the empty set, N_1 for the singleton set, N for the set of natural numbers, $\mathsf{List}(A)$ for the set of lists on the set A, $(\Sigma x \in A)\, B$ for the strong indexed sum, called here also dependent sum, of the family of sets $B\ set\ [x \in A]$ indexed on the set A, $A + B$ for the disjoint sum of the set A with the set B, $(\Pi x \in A)\, B$ for the dependent product set of the family of sets $B\ set\ [x \in A]$ indexed on the set A.

It is worth noting that the set N of natural numbers is not present in a primitive way in **mTT** since its rules can be derived by putting $\mathsf{N} \equiv \mathsf{List}(\mathsf{N}_1)$. Here, as in [24], we add it to the syntax of **mTT** because it plays a prominent role in the realizability interpretation in $\widehat{ID_1}$ and we want to avoid complications due to list encodings.

Finally, *collections* in **mTT** include the following types:

$$D\ col\ \ \equiv\ \ A\ set\ \mid\ \phi\ prop\ \mid\ \mathsf{prop_s}\ \mid\ A \rightarrow \mathsf{prop_s}\ \mid\ (\Sigma x \in D)\, E$$

where $\mathsf{prop_s}$ stands for the collection of small propositions and $A \rightarrow \mathsf{prop_s}$ for the collection of propositional functions of the set A, while $(\Sigma x \in D)\, E$ stands for the dependent sum of the collection family $E\ col\ [x \in D]$ indexed on the collection D. Collection constructors here are kept to a minimum in order to interpret power-collections of sets and contexts with dependent types which will be present in the

extensional level of **MF**.
All sets are collections thanks to the following rule:

$$\textbf{set-into-col)} \quad \frac{A\ set}{A\ col}$$

We end by mentioning the following relevant technical peculiarities of **mTT**:

- elimination rules of propositions act only toward propositions, as in CoC, to avoid the validity of choice principles contrary to what happens in Martin-Löf's type theory [1].

- in **mTT** we add explicitly substitution term equality rules of the form

$$\text{sub)} \quad \frac{\begin{array}{c} c \in C\,[\,x_1 \in A_1,\,\ldots,\,x_n \in A_n\,] \\ a_1 = b_1 \in A_1 \ \ldots\ a_n = b_n \in A_n[a_1/x_1, \ldots, a_{n-1}/x_{n-1}] \end{array}}{c[a_1/x_1, \ldots, a_n/x_n] = c[b_1/x_1, \ldots, b_n/x_n] \in C[a_1/x_1, \ldots, a_n/x_n]}$$

  in place of the usual term equality rules preserving term constructions typical of Martin-Löf's type theory MLtt in [31]. This choice yields a restriction of the valid equality rules in **mTT** with respect to those valid in MLtt. In particular in **mTT** the so called ξ-rule of lambda-terms

$$\xi\ \frac{c = c' \in C\ \ [x \in B]}{(\lambda x)c = (\lambda x)c' \in (\Pi x \in B)\,C}$$

  is not derivable.

It is worth recalling from [23] that the term equality rules of **mTT** are enough to interpret an extensional level including extensional equality of functions, as that represented by **emTT**, by means of the quotient model described in [23] and studied abstractly in [26, 25, 27].

mTT can be essentially viewed as a fragment of CoC by identifying collections with sets.

Moreover, **mTT** can be easily interpreted in intensional Martin-Löf's type theory MLtt in [31] by interpreting sets as MLtt-sets in the first universe and collections simply as MLtt-sets, propositions as sets according to the well-known isomorphism in [30] and the universe of small propositions as the first universe of MLtt.

[1] If you allow an elimination of existential quantifiers towards any type, you could build a function mapping a proof of an existential quantification $p \in (\exists x \in A)\,\phi$ towards the corresponding indexed sums $(\Sigma x \in A)\,\phi$ and by means of the first indexed sum projection you can extract a *choice function* whose value $f(p) \in A$ is a witness of the existential quantification.

2.3 The auxiliary type theory $\mathbf{mTT}^a$

Here we describe an auxiliary type theory, called $\mathbf{mTT}^a$, which is essentially an extension of **mTT** which we will validate in our categorical structure. The reason for interpreting $\mathbf{mTT}^a$, instead of simply **mTT**, is that the rules of $\mathbf{mTT}^a$ enjoy an easier proof of validity in our predicative variant of a realizability tripos.

First of all, in $\mathbf{mTT}^a$, as well as in the version of **mTT** interpreted in [24], the collection of small propositions $\mathsf{prop_s}$ is defined with codes à la Tarski as in [31], contrary to the version in [23], to make the interpretation easier to understand. Its rules are the following.
Elements of the collection of small propositions are generated as follows:

$$\mathrm{Pr}_1)\ \widehat{\bot} \in \mathsf{prop_s} \qquad \mathrm{Pr}_2)\ \frac{p \in \mathsf{prop_s} \quad q \in \mathsf{prop_s}}{p \,\widehat{\vee}\, q \in \mathsf{prop_s}}$$

$$\mathrm{Pr}_3)\ \frac{p \in \mathsf{prop_s} \qquad q \in \mathsf{prop_s}}{p \,\widehat{\rightarrow}\, q \in \mathsf{prop_s}} \qquad \mathrm{Pr}_4)\ \frac{p \in \mathsf{prop_s} \qquad q \in \mathsf{prop_s}}{p \,\widehat{\wedge}\, q \in \mathsf{prop_s}}$$

$$\mathrm{Pr}_5)\ \frac{A\ set \quad a \in A \quad b \in A}{\widehat{\mathsf{Eq}}(A, a, b) \in \mathsf{prop_s}} \qquad \mathrm{Pr}_6)\ \frac{p \in \mathsf{prop_s}\ [x \in A] \qquad A\ set}{\widehat{(\exists x \in A)}\, p \in \mathsf{prop_s}}$$

$$\mathrm{Pr}_7)\ \frac{p \in \mathsf{prop_s}\ [x \in A] \qquad A\ set}{\widehat{(\forall x \in A)}\, p \in \mathsf{prop_s}}$$

Elements of the collection of small propositions can be decoded as small propositions via an operator as follows

$$\tau\text{-Pr})\ \frac{p \in \mathsf{prop_s}}{\tau(p)\ prop_s}$$

and this operator satisfies the following definitional equalities:

$$\text{eq-Pr}_1)\ \tau(\widehat{\bot}) = \bot\, prop_s \qquad \text{eq-Pr}_2)\ \frac{p \in \mathsf{prop_s} \quad q \in \mathsf{prop_s}}{\tau(p \,\widehat{\vee}\, q) = \tau(p) \vee \tau(q)\, prop_s}$$

$$\text{eq-Pr}_3)\ \frac{p \in \mathsf{prop_s} \qquad q \in \mathsf{prop_s}}{\tau(p \,\widehat{\rightarrow}\, q) = \tau(p) \rightarrow \tau(q)\, prop_s} \qquad \text{eq-Pr}_4)\ \frac{p \in \mathsf{prop_s} \quad q \in \mathsf{prop_s}}{\tau(p \,\widehat{\wedge}\, q) = \tau(p) \wedge \tau(q)\, prop_s}$$

$$\text{eq-Pr}_5)\ \frac{A\ set \quad a \in A \quad b \in A}{\tau(\,\widehat{\mathsf{Eq}}(A, a, b)\,) = \mathsf{Eq}(A, a, b)\, prop_s}$$

$$\text{eq-Pr}_6)\ \frac{p \in \mathsf{prop_s}\ [x \in A] \qquad A\ set}{\tau(\widehat{(\exists x \in A)}\, p) = (\exists x \in A)\, \tau(p)\, prop_s}$$

$$\text{eq-Pr}_7)\ \frac{p \in \mathsf{prop_s}\ [x \in A] \qquad A\ set}{\tau(\widehat{(\forall x \in A)}\, p) = (\forall x \in A)\, \tau(p)\, prop_s}$$

Moreover, for the same reasons explained in [24] and essentially due to the need of interpreting the universe of small propositions in a clear way, even in $\mathbf{mTT}^a$ we add the collection Set of set codes whose related rules are the following. We do not add corresponding elimination and conversion rules as those of universes à la Tarski in [31] since they are not needed to prove the validity of **mTT**-rules.

Collection of sets

F-Se) $\mathsf{Set}\ col$

Elements of the collection of sets are generated as follows:

$$\text{sp-i-p)}\ \frac{p \in \mathsf{prop_s}}{\sigma(p) \in \mathsf{Set}} \qquad \text{Se}_e)\ \widehat{\mathsf{N}_0} \in \mathsf{Set}$$

$$\text{Se}_s)\ \widehat{\mathsf{N}_1} \in \mathsf{Set} \qquad \text{Se}_n)\ \widehat{\mathsf{N}} \in \mathsf{Set}$$

$$\text{Se}_l)\ \frac{a \in \mathsf{Set}}{\widehat{\mathsf{List}}(a) \in \mathsf{Set}} \qquad \text{Se}_u)\ \frac{a \in \mathsf{Set} \quad b \in \mathsf{Set}}{a \mathbin{\widehat{+}} b \in \mathsf{Set}}$$

$$\text{Se}_\Sigma)\ \frac{b \in \mathsf{Set}\ \ [x \in A] \qquad A\ set}{\widehat{(\Sigma x \in A)}\, b \in \mathsf{Set}} \qquad \text{Se}_\Pi)\ \frac{b \in \mathsf{Set}\ \ [x \in A] \qquad A\ set}{\widehat{(\Pi x \in A)}\, b \in \mathsf{Set}}$$

Set codes will be used to easily interpret the code of quantified small propositions.

Finally to further simplify the definition of the realizability interpretation, in $\mathbf{mTT}^a$ the elimination rules of some types, including disjoint sums, lists and natural numbers, are restricted to act toward *non-dependent types* and they are equipped with an extra equality rule expressing the uniqueness of the eliminator constructor as follows

2.3.1 Rules of disjoint sum

$$\text{+-f)}\ \frac{A\, set \qquad B\, set}{A + B\, set}$$

$$\text{+-i}_1)\ \frac{a \in A \qquad A\, set \qquad B\, set}{\mathsf{inl}(a) \in A + B} \qquad \text{+-i}_2)\ \frac{b \in B \qquad A\, set \qquad B\, set}{\mathsf{inr}(b) \in A + B}$$

$$\text{+-e)}\ \frac{c \in A + B \qquad C\, col \qquad d \in C\, [x \in A] \qquad e \in C\, [y \in B]}{\mathsf{El}_+(c, (x)\, d, (y)\, e) \in C}$$

$$\text{+-c}_1)\ \frac{a \in A \qquad C\, col \qquad d \in C\, [x \in A] \qquad e \in C\, [y \in B]}{\mathsf{El}_+(\mathsf{inl}(a), (x)\, d, (y)\, e) = d[a/x] \in C}$$

+-c$_2$) $$\frac{b \in B \qquad C\,col \qquad d \in C\,[x \in A] \qquad e \in C\,[y \in B]}{\mathsf{El}_{+}(\mathsf{inr}(b), (x)\,d, (y)\,e) = e[b/y] \in C}$$

+-η) $$\frac{p \in C + D \qquad t \in A\ [z \in C + D]}{\mathsf{El}_{+}(\,p, (x)\,t[\mathsf{inl}(x)/z]\,, (y)\,t[\mathsf{inr}(y)/z]\,) = t[p/z] \in A}$$

2.3.2 Rules of lists

List-f) $$\frac{A\,set}{\mathsf{List}(A)\,set}$$

List-i$_1$) $$\frac{A\,set}{\epsilon \in \mathsf{List}(A)}$$ List-i$_2$) $$\frac{A\,set \qquad b \in \mathsf{List}(A) \qquad a \in A}{\mathsf{cons}(b, a) \in \mathsf{List}(A)}$$

List-e) $$\frac{c \in \mathsf{List}(A) \qquad B\,col \qquad d \in B \qquad e \in B\,[x \in B, y \in A]}{\mathsf{El}_{\mathsf{List}}(c, d, (x, y)\,e) \in C}$$

List-c$_1$) $$\frac{B\,col \qquad d \in B \qquad e \in B\,[x \in B, y \in A]}{\mathsf{El}_{\mathsf{List}}(\epsilon, d, (x, y)\,e) = d \in C}$$

List-c$_2$) $$\frac{b \in \mathsf{List}(A) \qquad a \in A \qquad B\,col \qquad d \in B \qquad e \in B\,[x \in B, y \in A]}{\mathsf{El}_{\mathsf{List}}(\mathsf{cons}(b, a), d, (x, y)\,e) = e[\mathsf{El}_{\mathsf{List}}(b, d, (x, y)\,e)/x, a/y] \in C}$$

List-η) $$\frac{\begin{array}{l} B\,col \quad d \in B \quad e \in B\ [x \in B, y \in A] \quad t \in B\ [z \in \mathsf{List}(A)] \\ c \in \mathsf{List}(A) \qquad t[\epsilon/z] = a \in B \\ t[\mathsf{cons}(u, y)/z] = e[t[u/z]/x] \in B\ [u \in \mathsf{List}(A), y \in A] \end{array}}{\mathsf{El}_{\mathsf{List}}(c, d, (x, y)e) = t[c/z] \in L}$$

2.3.3 Rules of natural numbers set

N-f) $\mathsf{N}\,set$ N-i$_1$) $0 \in \mathsf{N}$ N-i$_2$) $$\frac{a \in \mathsf{N}}{\mathsf{succ}(a) \in \mathsf{N}}$$

N-e) $$\frac{a \in \mathsf{N} \quad A\,col \quad d \in A \quad e \in A\,[x \in A]}{\mathsf{El}_{\mathsf{N}}(a, d, (x)\,e) \in A}$$ N-c$_1$) $$\frac{A\,col \quad d \in A \quad e \in A\,[x \in A]}{\mathsf{El}_{\mathsf{N}}(0, d, (x)\,e) = d \in A}$$

$$\text{N-c}_2)\ \frac{a \in \mathsf{N} \qquad A\,col \qquad d \in A \qquad e \in A\,[x \in A]}{\mathsf{El_N}(\mathsf{succ}(a), d, (x)\,e) = e[\mathsf{El_N}(a, d, (x)\,e)/x] \in A}$$

$$\text{N-}\eta)\ \frac{\begin{array}{c} c \in \mathsf{N} \qquad A\,col \qquad t \in A\,[z \in \mathsf{N}] \qquad d \in A \qquad e \in A\,[x \in A] \\ t[0/z] = d \in A \qquad t[\mathsf{succ}(y)/z] = e[t[y/z]/x] \in A\,[y \in \mathsf{N}] \end{array}}{\mathsf{El_N}(c, d, (x)e) = t[c/z] \in A}$$

These rules do not change the expressive power of disjoint sums, lists and natural numbers. The reason is that, as first shown in [22], the above kinds of elimination rules with related equality rules are equivalent to the original ones of **mTT** provided that we add to $\mathbf{mTT}^a$ the following rules of extensional propositional equality of Martin-Löf's type theory in [30], which we also adopt in the extensional level of **MF** instead of those of the propositional identity Id:

$$\text{Eq-f})\ \frac{A\,col \qquad a \in A \qquad b \in A}{\mathsf{Eq}(A, a, b)\,prop} \qquad \text{Eq-f}_s)\ \frac{A\,set \qquad a \in A \qquad b \in A}{\mathsf{Eq}(A, a, b)\,prop_s}$$

$$\text{Eq-i})\ \frac{a \in A}{\mathsf{eq}(a) \in \mathsf{Eq}(A, a, a)} \qquad \text{Eq-e})\ \frac{p \in \mathsf{Eq}(A, a, b)}{a = b \in A} \qquad \text{Eq-}\eta)\ \frac{d \in \mathsf{Eq}(A, a, b)}{d = \mathsf{eq}(a) \in \mathsf{Eq}(A, a, b)}$$

and we add the usual equality rules preserving each type constructor as in [31, 30] or as those present in the extensional level of **MF** in [23].

Then we can equivalently define (see [31]) the strong indexed sums with the following rules

2.3.4 Rules of strong indexed sums

$$\Sigma\text{-f})\ \frac{A\,set \qquad B\,set\,[x \in A]}{(\Sigma x \in A)\,B\,set} \qquad \Sigma\text{-f}_{col})\ \frac{A\,col \qquad B\,col\,[x \in A]}{(\Sigma x \in A)\,B\,col}$$

$$\Sigma\text{-i})\ \frac{B\,col\,[x \in A] \qquad a \in A \qquad b \in B[a/x]}{\langle a, b\rangle \in (\Sigma x \in A)\,B}$$

$$\Sigma\text{-e}_1)\ \frac{c \in (\Sigma x \in A)\,B}{\pi_1(c) \in A} \qquad \Sigma\text{-e}_2)\ \frac{c \in (\Sigma x \in A)\,B}{\pi_2(c) \in B[\pi_1(c)/x]}$$

$$\Sigma\text{-c}_1)\ \frac{B\,col\,[x \in A] \qquad a \in A \qquad b \in B[a/x]}{\pi_1(\langle a, b\rangle) = a \in A};$$

$$\Sigma\text{-c}_2)\ \frac{B\,col\,[x \in A] \qquad a \in A \qquad b \in B[a/x]}{\pi_2(\langle a, b\rangle) = b \in B[a/x]}$$

$$\Sigma\text{-}\eta)\ \frac{c \in (\Sigma x \in A)\,B}{\langle \pi_1(c), \pi_2(c)\rangle = c \in (\Sigma x \in A)\,B}$$

Therefore we can easily show:

Proposition 2.1. *We can interpret* **mTT** *into* $\mathbf{mTT}^a$ *as the identity on all constructors except for those of the propositional equality* Id *which are interpreted as those of the extensional one* Eq*, and except for the strong indexed sum elimination constructor which is interpreted via projections.*

Proof. We briefly describe how to interpret the rules of **mTT**-strong indexed sums. Given $d \in (\Sigma x \in B)\,C$, $M\,col\,[\,z \in (\Sigma x \in B)\,C\,]$ and $m \in M[\langle x, y\rangle/z]\,[\,x \in B, y \in C\,]$ then

$$El_\Sigma(d, m) \ \equiv^{def}\ m[\pi_1(d)/x, \pi_2(d)/y]$$

is of type $M[\langle \pi_1(d), \pi_2(d)\rangle/z]$ by definition. But by the substitution rules and the rule conv) [2] (see the rules of **mTT** in [23]) and the above Σ-η of $\mathbf{mTT}^a$ we conclude that it is of type $M(d)$ as well, as required.

Concerning the propositional equality: the constructor $\mathsf{id}_A(a)$ of **mTT** is interpreted as $\mathsf{eq}(a)$ of $\mathbf{mTT}^a$ and the elimination constructor $El_{\mathsf{Id}}(p, (x)c)$ as $c[a/x]$, given that its type $C(a, a, \mathsf{eq}(a))$ happens to be equal to $C(a, b, p)$ by the rules subT) and conv) in [23] since from $p \in \mathsf{Eq}(A, a, b)$ we get $a = b \in A$ and also $p = \mathsf{eq}(a) \in \mathsf{Eq}(A, a, b)$ by the rules of Eq. □

3 Feferman's theory of inductive definitions $\widehat{ID_1}$

The system $\widehat{ID_1}$ is a predicative fragment of second-order arithmetic, more precisely it is the predicative fragment of second-order arithmetic extending Peano arithmetic with some (not necessarily least) fixpoints for each positive arithmetical operator. Its number terms are number variables (or simply variables) $\xi_1, ..., \xi_n ...$, the constant 0 and the terms built by applying the unary successor functional symbol *succ* and the binary sum and product functional symbols $+$ and $*$ to number terms. Set terms are only set variables $X, Y, Z....$ The *arithmetical* formulas are obtained starting from $t = s$ and $t\,\varepsilon\,X$ with t, s number terms and X a set variable, by applying the connectives $\wedge$, $\vee$, $\neg$, $\rightarrow$ and the number quantifiers $\forall x$, $\exists x$. Moreover let us give the following two definitions.

[2]We just recall that this rule says that from $a \in A$ and $A = B$ *type* we get $a \in B$.

Definition 3.1. *An occurrence of a set variable X in an arithmetical formula φ is* positive *or* negative *according to the following conditions.*

1. *the occurrence of X in $t \,\varepsilon\, X$, where t is a number term, is positive;*
2. *a positive (negative) occurrence of X in ψ, is positive (negative) in $\psi \wedge \phi$, $\phi \wedge \psi$, $\phi \vee \psi$, $\psi \vee \phi$, $\phi \to \psi$, $\exists x\, \psi$ and $\forall x\, \psi$;*
3. *a positive (negative) occurrence of X in ψ, is negative (positive) in $\psi \to \phi$ and $\neg\, \psi$.*

Definition 3.2. *An arithmetical formula φ with exactly one free number variable x and one free set variable X which occurs only positively is called an* admissible *formula.*

In order to define the system $\widehat{ID_1}$ we add to the language of second-order arithmetic a unary predicate symbol P_φ for every admissible formula φ . The atomic formulas of $\widehat{ID_1}$ are

1. $t = s$ with t and s number terms;
2. $t \,\varepsilon\, X$ with t a number term and X a set variable;
3. $P_\varphi(t)$ with t a number term and φ an admissible formula.

All formulas of $\widehat{ID_1}$ are obtained from atomic formulas by applying connectives, number quantifiers and set quantifiers.

The axioms of $\widehat{ID_1}$ are the axioms of Peano Arithmetic plus the following three axiom schemata:

1. *Comprehension schema*: for all formulas $\varphi(x)$ of $\widehat{ID_1}$ without set quantifiers
$$\exists X\, \forall x\, (x \,\varepsilon\, X \leftrightarrow \varphi(x))$$
provided that X is not free in $\varphi(x)$
2. *Induction schema*: for all formulas $\varphi(x)$ of $\widehat{ID_1}$ without set quantifiers
$$(\varphi(0)\, \wedge\, \forall x\, (\varphi(x) \to \varphi(succ(x)))) \to \forall x\, \varphi(x)$$
3. *Fixpoint schema*: for all admissible formulas φ with x and X free
$$\varphi[P_\varphi/X] \leftrightarrow P_\varphi(x)$$
where $\varphi[P_\varphi/X]$ is the result of substituting in φ every atomic subformula $t \,\varepsilon\, X$ with $P_\varphi(t)$.

The system $\widehat{ID_1}$ allows us to define predicates as fixpoints, by using axiom schema 3, if they are presented in an appropriate way (i. e. using admissible formulas).

3.1 Notations of recursive functions in $\widehat{ID_1}$

A *numeral* is a term of the form $succ(succ...succ(0))$. As usual we denote numerals with boldface lower case letters $\mathbf{n}$.

In $\widehat{ID_1}$ one can certainly represent a Gödelian coding of recursive functions by means of the Kleene predicate $T(x, y, z)$ and the primitive recursive (meta)function U. First of all we define *applicative terms* as follows (notice that these terms are not part of the syntax of $\widehat{ID_1}$, but are auxiliary terms):

1. every number variable is an applicative term;
2. every numeral is an applicative term;
3. if t and s are applicative terms, then $\{t\}(s)$ is an applicative term.

We use the abbreviation $\{s\}(t_1, ..., t_n)$ for applicative terms $s, t_1, ..., t_n, ...$ as follows

1. $\{s\}()$ is s;
2. $\{s\}(t_1, ..., t_{n+1})$ is $\{\{s\}(t_1, ..., t_n)\}(t_{n+1})$.

If $\varphi(\overline{x}, x)$ is a formula of $\widehat{ID_1}$ and t is an applicative term, then we define $\varphi(\overline{x}, t)$ by induction on the definition of applicative terms t for all formulas as follows:

1. $\varphi(\overline{x}, y)$ is itself;
2. if $\mathbf{n}$ is a numeral, $\varphi(\overline{x}, \mathbf{n})$ is itself;
3. $\varphi(\overline{x}, \{t\}(s))$ is $\exists x(T(t, s, x) \wedge \varphi(\overline{x}, U(x)))$.

Notice that if $\{t\}(s)$ is an applicative term, the formula $\{t\}(s) = \{t\}(s)$ turns out to be equivalent to what is usually denoted with $\{t\}(s)\downarrow$ i. e. the formula $\exists x\, T(t, s, x)$. In particular, for a *generic* applicative term t it can be proved that the formula $t = t$ is provable when *the applicative term t converges*. Hence it makes sense to introduce the formula $t \simeq s$ as an abbreviation for $t = t \vee s = s \rightarrow t = s$ for every pair of *generic* applicative terms t and s.

If t is an applicative term with all variables among $x_1, ..., x_n$, then there is a numeral $\Lambda x_1...\Lambda x_n.t$ for which

$$\widehat{ID_1} \vdash \forall x_1...\forall x_n(\{\Lambda x_1...\Lambda x_n.t\}(x_1, ..., x_n) \simeq t)$$

For $1 \leq j \leq n$ we define a numeral π^n_j as $\Lambda x_1....\Lambda x_n.x_j$. These numerals obviously satisfy the following

$$\widehat{ID_1} \vdash \{\pi^n_j\}(x_1, ..., x_n) = x_j$$

Any n-ary primitive recursive (meta)function f can be represented by a numeral $\mathbf{f}$ through the Gödelian coding in such a way that

$$\widehat{ID_1} \vdash \{\mathbf{f}\}(x_1, ..., x_n) = f(x_1, ..., x_n)$$

In particular there exist numerals $\mathbf{p}, \mathbf{p}_1, \mathbf{p}_2$ and $\mathbf{s}$ representing a primitive recursive pairing function p with primitive recursive projections p_1, p_2 and the successor function.

We define for $1 \leq j \leq n$, numerals $\mathbf{p}^n$ and $\mathbf{p}^n_j$, representing the encoding of n-tuples of natural numbers and the relative jth projections as follows:

1. $\mathbf{p}^1$ and $\mathbf{p}^1_1$ are both π^1_1;

2. $\mathbf{p}^{n+1}$ is $\Lambda x_1...\Lambda x_{n+1}.\{\mathbf{p}\}(\{\mathbf{p}^n\}(x_1, ..., x_n), x_{n+1})$;

3. $\mathbf{p}^{n+1}_j$ is $\Lambda x.\{\mathbf{p}^n_j\}(\{\mathbf{p}_1\}(x))$ if $1 \leq j \leq n$;

4. $\mathbf{p}^{n+1}_{n+1}$ is $\mathbf{p}_2$.

We have that for $n \geq 1$

1. $\widehat{ID_1} \vdash \{\mathbf{p}^n\}(\{\mathbf{p}^n_1\}(x), ..., \{\mathbf{p}^n_n\}(x)) = x$

2. $\widehat{ID_1} \vdash \{\mathbf{p}^n_j\}(\{\mathbf{p}^n\}(x_1, ..., x_n)) = x_j$ for every $1 \leq j \leq n$.

We can bijectively encode finite lists of natural numbers $[n_0, ..., n_k]$ with natural numbers in such a way that the component functions $(\ )_j$, the length function $lh(\ )$ and the concatenation function cnc of lists with natural numbers are primitive recursive and that the empty list is coded by 0. In particular there exists a numeral $\mathbf{cnc}$ for which $\widehat{ID_1} \vdash \{\mathbf{cnc}\}(x, y) = cnc(x, y)$.

Moreover there exists a list recursor, i.e. a numeral $\mathbf{listrec}$ for which

1. $\widehat{ID_1} \vdash \{\mathbf{listrec}\}(0, y, z) \simeq y$

2. $\widehat{ID_1} \vdash \{\mathbf{listrec}\}(\{\mathbf{cnc}\}(x, x'), y, z) \simeq \{z\}(\{\mathbf{listrec}\}(x, y, z), x')$

4 The effective pretripos for mTT

In this section we are going to define in $\widehat{ID_1}$ a predicative categorical structure, called *effective pretripos for* **mTT**, which represents a predicative variant of a realizability tripos validating **mTT**. In a broad sense it can be considered a predicative variant of the effective tripos giving rise to Hyland's effective topos Eff in [17]. Indeed, our ultimate goal is to use our effective pretripos to build a predicative variant of a realizability topos like Eff.

Recall from [18, 33] that a tripos is an indexed category

$$P : \mathcal{C}^{op} \longrightarrow \mathbf{Cat}$$

which is a Lawvere-first order hyperdoctrine in the category of Heyting algebras enriched with a weak subobject classifier, called a generic predicate in [18], capable of producing power-sets in the category obtained by applying the so called tripos-to-topos construction. This weak classifier is of an impredicative nature and it must be necessarily so.

Here we are going to define a predicative variant of a tripos with the idea of getting just power-collections and not power-sets in the corresponding predicative variant of the tripos-to-topos construction. These will be structured in a fully analogous way to the two-level structure of **MF** where the universes of small propositions and of propositional functions on any set at the intensional level of **MF** are enough to model power-collections of sets at the extensional level of **MF** by means of a quotient model (see [23]).

We now briefly outline the categorical structure of our predicative variant of a realizability tripos by describing what we are going to include in it:

- We define an indexed category of "realized" propositions

  $$\mathbf{Prop} : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

  on a category **Cont** of "realized contexts" and realized morphisms between them, equipped with the structure of a Lawvere's first order hyperdoctrine but in the category of Heyting prealgebras [3]. The category **Cont** will host a realizability interpretation of $\mathbf{mTT}^a$-contexts as that in [24]. This category is also equivalent to its full subcategory $\mathcal{C}$ of realized collections, which are defined as subsets of natural numbers in $\widehat{ID_1}$ equipped with an equivalence relation, whose morphisms turn out to be suitable recursive operations. Each

[3] A Heyting prealgebra is a preorder whose posetal reflection is a Heyting algebra.

fibre of **Prop** represents the category of realized propositions defined in a proof-irrelevant way as subsets of a singleton.

We use the category of contexts **Cont** instead of $\mathcal{C}$ as the base of our categorical structure, because the realizability interpretation of $\mathbf{mTT}^a$-contexts and generic $\mathbf{mTT}^a$-judgements becomes simpler.

It is worth noting that the category **Cont** has also an indexed structure of families of realized collections

$$\mathbf{Col} : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

whose fibre on the empty context $[\,]$ is equivalent to **Cont**. Moreover, it contains **Prop** as a sub-indexed category

$$\mathbf{Prop} \hookrightarrow \mathbf{Col}$$

- We define a realized collection US via a fixpoint formula of $\widehat{ID_1}$, which will host the realizability interpretation of the collection of $\mathbf{mTT}^a$-sets. This is defined as in [24] following a technique due to Beeson [8].

  US is crucial to define the (indexed) category of families of realized sets

  $$\mathbf{Set} : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

  which is a sub-indexed structure of **Col**

  $$\mathbf{Set} \hookrightarrow \mathbf{Col}$$

  Namely families of realized sets are families of realized collections classified by the non-dependent realized collection US, in the sense that US represents the indexed functor **Set** via a natural bijection

  $$\mathbf{Set}(\Gamma) \simeq \mathbf{Cont}(\Gamma, \mathsf{US})$$

  for objects Γ in **Cont**.

- We define a realized collection USP as a sub-collection of US, which will host the realizability interpretation of the collection of $\mathbf{mTT}^a$-small propositions. This is also defined as in [24].

  The construction of USP is crucial to define the first-order hyperdoctrine of realized small propositions

  $$\mathbf{Prop}_s : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

which is a subindexed category both of **Prop** and of **Set**

$$\begin{tikzcd} \mathbf{Set} \arrow[r, hook] & \mathbf{Col} \\ \mathbf{Prop}_s \arrow[u, hook] \arrow[r, hook] & \mathbf{Prop} \arrow[u, hook] \end{tikzcd}$$

and is classified by USP in the sense that USP represents the indexed functor $\mathbf{Prop}_s$ via a natural bijection for objects Γ in **Cont**

$$\mathbf{Prop}_s(\Gamma) \simeq \mathbf{Cont}(\Gamma, \mathsf{USP})$$

This classification property provides an *intensional predicative* version of the original weak subobject classifier property of a tripos.

In the next sections we will often include lemmas and theorems without proofs because their proofs just involve straightforward verifications.

4.1 The category of realized collections in $\widehat{ID_1}$

Here we are going to define the category of *realized collections*. We will denote such a category as $\mathcal{C}$.

A realized collection will denote a quotient of a subset of natural numbers acting as realizers. It is represented in $\widehat{ID_1}$ by a first-order formula defining the realizers together with an equivalence relation $x \sim y$. Morphisms between realized collections will be defined as recursive functions between them preserving the corresponding equivalence relations and called *recursive operations*.

We start by giving the notion of dependent realized collection, namely a family of realized collections depending on a finite number of variables. From this notion we will deduce that of realized collection.

Definition 4.1. *Let $\overline{x}$ be a (possibly empty) list of distinct variables of the language of $\widehat{ID_1}$. A* realized collection of $\widehat{ID_1}$ depending on $\overline{x}$ *(or simply a dependent realized collection) is a pair $A(\overline{x}) := (|A(\overline{x})|, x \sim_{A(\overline{x})} y)$ where*

1. *$|A(\overline{x})|$ is a first-order definable class of $\widehat{ID_1}$, i. e. it is a formal expression*

$$\{x \,|\, \phi_A(\overline{x}, x)\}$$

where x is a variable different from those in $\overline{x}$ and $\phi_A(\overline{x}, x)$ is a first-order formula of $\widehat{ID_1}$, namely a formula without set variables and set quantifiers, but possibly with fixpoint predicates P_φ, with all free variables among those in $\overline{x}$

and x*. We will write* $x \,\varepsilon\, A(\overline{x})$ *as an abbreviation for* $\phi_A(\overline{x}, x)$*, since we may think of* $A(\overline{x})$ *as a subset* $|A(\overline{x})|$ *of natural numbers, called* realizers*, equipped with a relation* $\sim_{A(\overline{x})}$*.*

2. $x \sim_{A(\overline{x})} y$ *is a first-order definable equivalence relation on* $|A(\overline{x})|$*, i. e. it is a first-order formula of* $\widehat{ID_1}$*, where* x *and* y *are distinct variables and they are different from those in* $\overline{x}$*, with all free variables among those in* $\overline{x}$*,* x *or* y *for which:*
   (a) $x \sim_{A(\overline{x})} y \vdash_{\widehat{ID_1}} x \,\varepsilon\, A(\overline{x}) \wedge y \,\varepsilon\, A(\overline{x})$
   (b) $x \,\varepsilon\, A(\overline{x}) \vdash_{\widehat{ID_1}} x \sim_{A(\overline{x})} x$
   (c) $x \sim_{A(\overline{x})} y \vdash_{\widehat{ID_1}} y \sim_{A(\overline{x})} x$
   (d) $x \sim_{A(\overline{x})} y \wedge y \sim_{A(\overline{x})} z \vdash_{\widehat{ID_1}} x \sim_{A(\overline{x})} z$

We identify dependent realized collections $A(\overline{x})$ *and* $B(\overline{x})$ *for which*

$$\widehat{ID_1} \vdash x \sim_{A(\overline{x})} y \leftrightarrow x \sim_{B(\overline{x})} y$$

(this automatically ensures that $\widehat{ID_1} \vdash x \,\varepsilon\, A(\overline{x}) \leftrightarrow x \,\varepsilon\, B(\overline{x})$*, namely the validity of subset extensional equality).*

Definition 4.2. *A* realized collection *of* $\widehat{ID_1}$ *is a realized collection depending on the empty list.*

Definition 4.3. *Given two realized collections* A *and* B*, a* recursive operation *(or simply an operation) from* A *to* B *is an equivalence class* $[\mathbf{n}]_{\approx_{A,B}}$ *of numerals for which*

$$x \sim_A y \vdash_{\widehat{ID_1}} \{\mathbf{n}\}(x) \sim_B \{\mathbf{n}\}(y)$$

with respect to the equivalence relation given by

$$\mathbf{n} \approx_{A,B} \mathbf{m} \text{ if and only if } x \,\varepsilon\, A \vdash_{\widehat{ID_1}} \{\mathbf{n}\}(x) \sim_B \{\mathbf{m}\}(x)$$

Definition 4.4. *We call* $\mathcal{C}$ *the category of realized collections of* $\widehat{ID_1}$ *and recursive operations between them where the composition of morphisms and identities are defined as follows.*

If $[\mathbf{n}]_{\approx_{A,B}}$ *is an operation from a realized collection* A *to a realized collection* B *and* $[\mathbf{m}]_{\approx_{B,C}}$ *is an operation from a realized collection* B *to a realized collection* C*, then their composition is the operation*

$$[\mathbf{m}]_{\approx_{B,C}} \circ [\mathbf{n}]_{\approx_{A,B}} := [\,\Lambda x.\{\mathbf{m}\}(\{\mathbf{n}\}(x))\,]_{\approx_{A,C}}$$

If A *is a realized collection, then its identity* id_A *is defined as* $[\pi^1_1]_{\approx_{A,A}}$*.*

4.2 The category of realized contexts in $\widehat{ID_1}$

Here we are going to define the category **Cont** of *realized contexts* and *realized morphisms* between them. This category will be used to interpret the telescopic contexts of dependent types of $\mathbf{mTT}^a$. We will deduce the categorical properties which are necessary to validate $\mathbf{mTT}^a$ from those of the category of realized collections $\mathcal{C}$, being **Cont** equivalent to $\mathcal{C}$. Indeed the categorical structure of $\mathcal{C}$ will be easier to describe.

We start by giving some abbreviations on list of variables.

Fix two countable sequences of variables $x_1, ..., x_n...$ and $y_1, ..., y_n...$ in such a way that all these variables are distinct. We denote by $\overline{x}_{|_j}$ the empty list if $j = 0$ or the list $x_1, ..., x_j$ otherwise. Similarly we define $\overline{y}_{|_j}$.

Then, we use the abbreviation $\Lambda\overline{x}_{|_j}$ for $\Lambda x_1 ... \Lambda x_j$ if $j > 0$, while $\Lambda\overline{x}_{|_0}$ means no Λ-quantification. In case of an empty list of variables $A(\,)$ means A.

If $\overline{\mathbf{k}}$ is a finite list of numerals with length n, then for $j \leq n$, we use the abbreviation $\{\overline{\mathbf{k}}_{|_j}\}(\overline{t})$ for the empty list if $j = 0$, while $\{\overline{\mathbf{k}}_{|_j}\}(\overline{t})$ is the list $\{\mathbf{k}_1\}(\overline{t}), ..., \{\mathbf{k}_j\}(\overline{t})$ otherwise; we write $\{\overline{\mathbf{k}}\}(\overline{t})$ as an abbreviation for $\{\overline{\mathbf{k}}_{|_n}\}(\overline{t})$.

Definition 4.5. *A* realized context *(or simply a context) of $\widehat{ID_1}$ is a (possibly empty) finite list*

$$\Gamma = [A_1, ..., A_j(\overline{x}_{|_{j-1}}), ..., A_n(\overline{x}_{|_{n-1}})]$$

where $A_j(\overline{x}_{|_{j-1}})$ is a collection of $\widehat{ID_1}$ depending on $\overline{x}_{|_{j-1}}$ for $1 \leq j \leq n$, which satisfies the following conditions:

1. $x_{j+1} \sim_{A_{j+1}(\overline{x}_{|_j})} y_{j+1} \vdash_{\widehat{ID_1}} x_1 \,\varepsilon\, A_1 \,\wedge\, ... \,\wedge\, x_j \,\varepsilon\, A_j(\overline{x}_{|_{j-1}})$

2. $x_1 \sim_{A_1} y_1 \,\wedge\, ... \,\wedge\, x_j \sim_{A_j(\overline{x}_{|_{j-1}})} y_j \vdash_{\widehat{ID_1}}$

$$x_{j+1} \sim_{A_{j+1}(\overline{x}_{|_j})} y_{j+1} \leftrightarrow x_{j+1} \sim_{A_{j+1}(\overline{y}_{|_j})} y_{j+1}$$

for every $1 \leq j \leq n-1$.

Moreover, for a realized context Γ of $\widehat{ID_1}$, the length $\ell(\Gamma)$ of Γ is the length of Γ as a list.

Finally, if $\Gamma = [A_1, ..., A_n(\overline{x}_{|_{n-1}})]$ is a realized context of $\widehat{ID_1}$ with positive length n, then

1. $\overline{x}_{|_n} \,\varepsilon\, \Gamma$ *is an abbreviation for* $x_1 \,\varepsilon\, A_1 \,\wedge ... \wedge\, x_n \,\varepsilon\, A_n(\overline{x}_{|_{n-1}})$

2. $\overline{x}_{|_n} \sim_\Gamma \overline{y}_{|_n}$ *is an abbreviation for* $x_1 \sim_{A_1} y_1 \wedge ... \wedge x_n \sim_{A_n(\overline{x}_{|_{n-1}})} y_n$

If Γ is the empty list, then $\overline{x}_{|_0}\,\varepsilon\,\Gamma$ and $\overline{y}_{|_0} \sim_\Gamma \overline{y}_{|_0}$ are both the true constant $\top$.

Definition 4.6. *If Γ and Γ' are contexts of $\widehat{ID_1}$, then a* realized morphism *from Γ to Γ' is an equivalence class $[\overline{\mathbf{k}}]_{\approx_{\Gamma,\Gamma'}}$ of lists of numerals with length equal to the length of Γ' satisfying the following requirements: if $\Gamma' = [B_1, ..., B_n(\overline{x}_{|_{n-1}})]$ with $n > 0$, then for all $1 \leq j \leq n$:*

$$\overline{x}_{|_{\ell(\Gamma)}} \sim_\Gamma \overline{y}_{|_{\ell(\Gamma)}} \vdash_{\widehat{ID_1}} \{\mathbf{k}_j\}(\overline{x}_{|_{\ell(\Gamma)}}) \sim_{B_j(\{\overline{\mathbf{k}}_{|_{j-1}}\}(\overline{x}_{|_{\ell(\Gamma)}}))} \{\mathbf{k}_j\}(\overline{y}_{|_{\ell(\Gamma)}})$$

with respect to the equivalence relation $\approx_{\Gamma,\Gamma'}$ defined by $\overline{\mathbf{k}} \approx_{\Gamma,\Gamma'} \overline{\mathbf{k}'}$ if and only if

$$\overline{x}_{|_{\ell(\Gamma)}}\,\varepsilon\,\Gamma \vdash_{\widehat{ID_1}} \{\mathbf{k}_j\}(\overline{x}_{|_{\ell(\Gamma)}}) \sim_{B_j(\{\overline{\mathbf{k}}_{|_{j-1}}\}(\overline{x}_{|_{\ell(\Gamma)}}))} \{\mathbf{k}'_j\}(\overline{x}_{|_{\ell(\Gamma)}})$$

for every $1 \leq j \leq n$.

In the case in which $\Gamma' = [\,]$, then the unique realized morphism is the class $!_{\Gamma,[\,]} := [\,]_{\approx_{\Gamma,[\,]}}$ containing only the empty list.

Definition 4.7. *If $\overline{\mathbf{k}}$ and $\overline{\mathbf{h}}$ are lists of numerals and n is a natural (meta)number, then*

$$\overline{\mathbf{h}} \circ^n \overline{\mathbf{k}} := [\,\Lambda\overline{x}_{|_n}.\{\mathbf{h}_1\}(\{\overline{\mathbf{k}}\}(\overline{x}_{|_n})), ..., \Lambda\overline{x}_{|_n}.\{\mathbf{h}_{\ell(\overline{\mathbf{h}})}\}(\{\overline{\mathbf{k}}\}(\overline{x}_{|_n}))\,]$$

Definition 4.8. *If $[\overline{\mathbf{k}}]_{\approx_{\Gamma,\Gamma'}} : \Gamma \to \Gamma'$ and $[\overline{\mathbf{h}}]_{\approx_{\Gamma',\Gamma''}} : \Gamma' \to \Gamma''$ are realized morphisms between contexts of $\widehat{ID_1}$, then we define their composition as the realized morphism*

$$[\overline{\mathbf{h}}]_{\approx_{\Gamma',\Gamma''}} \circ [\overline{\mathbf{k}}]_{\approx_{\Gamma,\Gamma'}} := [\,\overline{\mathbf{h}} \circ^{\ell(\Gamma)} \overline{\mathbf{k}}\,]_{\approx_{\Gamma,\Gamma''}} : \Gamma \to \Gamma''$$

If Γ is a context of $\widehat{ID_1}$, then its identity is defined as the realized morphism

$$[\,\pi_1^{\ell(\Gamma)}, ..., \pi_{\ell(\Gamma)}^{\ell(\Gamma)}\,]_{\approx_{\Gamma,\Gamma}} : \Gamma \to \Gamma$$

if $\ell(\Gamma) > 0$, while it is the realized morphism

$$[\,]_{\approx_{[\,],[\,]}} : [\,] \to [\,]$$

if $\Gamma = [\,]$.

Theorem 4.9. *Realized contexts of $\widehat{ID_1}$ and realized morphisms between them with their compositions and identities form a category denoted by* **Cont**.

As it happens in dependent type theory contexts can be equivalently represented as the indexed sums of their components. To this purpose we define the following realized morphisms which will act as projections to extract the components of a context:

Definition 4.10. *If* Γ *is a context of* $\widehat{ID_1}$ *and* n *is a natural (meta)number, we define the realized morphisms* pr_Γ *and* $\mathsf{pr}_\Gamma^{(n)}$ *in* **Cont** *as follows:*

- $\mathsf{pr}_{[\,]}$ *is* $id_{[\,]}$ *and* $\mathsf{pr}_{[A]}$ *is* $[\,]_\approx : [A] \to [\,]$;[4]

  $\mathsf{pr}_{[\Gamma,A]}$ *is* $[\pi_1^{\ell(\Gamma)+1}, ..., \pi_{\ell(\Gamma)}^{\ell(\Gamma)+1}]_\approx : [\Gamma, A] \to \Gamma$ *if* $\ell(\Gamma) > 0$;
- pr_Γ^0 *is* id_Γ *and* $\mathsf{pr}_\Gamma^{(i+1)}$ *is* $\mathsf{pr}_{\mathsf{cod}(\mathsf{pr}_\Gamma^{(i)})} \circ \mathsf{pr}_\Gamma^{(i)}$

where $\mathsf{cod}(\mathsf{pr}_\Gamma^{(i)})$ *denotes the codomain of* $\mathsf{pr}_\Gamma^{(i)}$.

Now we define the indexed sum of the last two components of a context:

Definition 4.11. *Suppose* $[\Gamma, A(\overline{x}_{|_{\ell(\Gamma)}}), B(\overline{x}_{|_{\ell(\Gamma)+1}})]$ *is a realized context of* $\widehat{ID_1}$. *We define the* indexed sum collection

$$\Sigma^\Gamma(\, A(\overline{x}_{|_{\ell(\Gamma)}}),\, B(\overline{x}_{|_{\ell(\Gamma)+1}})\,)$$

as a collection depending on $\overline{x}_{|_{\ell(\Gamma)}}$ *determined by the following conditions:*[5]

$$x \,\varepsilon\, \Sigma^\Gamma(\, A(\overline{x}_{|_{\ell(\Gamma)}}),\, B(\overline{x}_{|_{\ell(\Gamma)+1}})\,) \equiv^{def} p_1(x) \,\varepsilon\, A(\overline{x}_{|_{\ell(\Gamma)}}) \,\wedge\, p_2(x) \,\varepsilon\, B(\overline{x}_{|_{\ell(\Gamma)}}, p_1(x))$$

$$x \sim_{\Sigma^\Gamma(\, A(\overline{x}_{|_{\ell(\Gamma)}}), B(\overline{x}_{|_{\ell(\Gamma)+1}})\,)} y \equiv^{def} p_1(x) \sim_{A(\overline{x}_{|_{\ell(\Gamma)}})} p_1(y) \,\wedge\, p_2(x) \sim_{B(\overline{x}_{|_{\ell(\Gamma)}}, p_1(x))} p_2(y)$$

Clearly, the indexed sum collection allows to represent a context in an equivalent way as follows:

Lemma 4.12. *Suppose* $[\Gamma, A(\overline{x}_{|_{\ell(\Gamma)}}), B(\overline{x}_{|_{\ell(\Gamma)+1}})]$ *is a realized context of* $\widehat{ID_1}$. *Then,* $[\Gamma, A(\overline{x}_{|_{\ell(\Gamma)}}), B(\overline{x}_{|_{\ell(\Gamma)+1}})]$ *is isomorphic to* $[\Gamma, \Sigma^\Gamma(A(\overline{x}_{|_{\ell(\Gamma)}}), B(\overline{x}_{|_{\ell(\Gamma)+1}}))]$ *in* **Cont**.

Proof. If Γ is not empty, just take the realized morphism from $[\Gamma, A(\overline{x}_{|_{\ell(\Gamma)}}), B(\overline{x}_{|_{\ell(\Gamma)+1}})]$ to $[\Gamma, \Sigma^\Gamma(\, A(\overline{x}_{|_{\ell(\Gamma)}}),\, B(\overline{x}_{|_{\ell(\Gamma)+1}})\,)]$ determined by the list

$$[\, \pi_1^{\ell(\Gamma)+2}, ..., \pi_{\ell(\Gamma)}^{\ell(\Gamma)+2}, \Lambda\overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\Lambda y.\{\mathbf{p}\}(x,y)\,]$$

Its inverse from $[\Gamma, \Sigma^\Gamma(\, A(\overline{x}_{|_{\ell(\Gamma)}}),\, B(\overline{x}_{|_{\ell(\Gamma)+1}})\,)]$ to $[\Gamma, A(\overline{x}_{|_{\ell(\Gamma)}}), B(\overline{x}_{|_{\ell(\Gamma)+1}})]$ is the realized morphism determined by the list

$$[\, \pi_1^{\ell(\Gamma)+1}, ..., \pi_{\ell(\Gamma)}^{\ell(\Gamma)+1}, \Lambda\overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}_1\}(x), \Lambda\overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}_2\}(x)\,]$$

Instead, if Γ is empty, we can consider the realized isomorphism determined by $[\mathbf{p}]$ and $[\mathbf{p}_1, \mathbf{p}_2]$. ☐

[4] The subscripts on $\approx$ will be omitted when they will be clear from the context

[5] Here as usual x and y are fresh distinct variables

Theorem 4.13. **Cont** *is equivalent to* $\mathcal{C}$*.*

Proof. From $\mathcal{C}$ to **Cont** take the functor **F** sending any collection A of $\mathcal{C}$ to $[A]$ and every realized morphism $[\mathbf{n}]_{\approx_{A,B}}$ to $[\mathbf{n}]_{\approx_{[A],[B]}}$. Then, define a functor **E** from **Cont** to $\mathcal{C}$ as follows:

1. $\mathbf{E}([\,])$ is $(\ \{x\,|\,x=0\}\ ,\ \ x=y\ \wedge\ x=0\ )$,

   $\mathbf{E}([A])$ is A,

   $\mathbf{E}([\Gamma, A])$ is $\Sigma^{[\,]}(\ \mathbf{E}(\Gamma)\ ,\ A(\{\mathbf{p}_1^{\ell(\Gamma)}\}(x_1), ..., \{\mathbf{p}_{\ell(\Gamma)}^{\ell(\Gamma)}\}(x_1))\ )$;

2. if $\ell(\Gamma) > 0$ and $\ell(\Gamma') > 0$, then $[\overline{\mathbf{k}}]_{\approx_{\Gamma,\Gamma'}} : \Gamma \to \Gamma'$ is sent to

   $$\Big[\,\Lambda x.\{\mathbf{p}^{\ell(\Gamma')}\}(\{\overline{\mathbf{k}}\}(\{\mathbf{p}_1^{\ell(\Gamma)}\}(x), ..., \{\mathbf{p}_{\ell(\Gamma)}^{\ell(\Gamma)}\}(x)))\,\Big]_{\approx_{\mathbf{E}(\Gamma),\mathbf{E}(\Gamma')}}$$

   if $\Gamma = [\,]$ and $\ell(\Gamma') > 0$, then $[\overline{\mathbf{k}}]_{\approx_{[\,],\Gamma'}} : [\,] \to \Gamma'$ is sent to

   $$[\ \ \Lambda x.\{\mathbf{p}^{\ell(\Gamma')}\}(\overline{\mathbf{k}})\ \ ]_{\approx_{\mathbf{E}([\,]),\mathbf{E}(\Gamma')}}$$

   and $!_{\Gamma,[\,]} : \Gamma \to [\,]$ is sent to $[\,\Lambda x.0\,]_{\approx_{\mathbf{E}(\Gamma),\mathbf{E}([\,])}}$.

□

4.3 Families of realized collections as an indexed category

Here we are going to define an indexed category on the category of realized contexts

$$\mathbf{Col} : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

whose fibre on a context Γ will be defined as a presentation of the slice category $\mathbf{Cont}/\Gamma$ in terms of *families of realized collections depending on the context* Γ.

Definition 4.14. *If* Γ *is a context of* $\widehat{ID_1}$ *(i. e. an object of* **Cont***), then* A *is a* family of realized collections *on* Γ *(or a* realized collection depending on Γ*) if and only if* $[\Gamma, A]$ *is a context of* $\widehat{ID_1}$*.*

Definition 4.15. *Let* $\mathbf{Col}(\Gamma)$ *be the category whose objects are families of realized collections on* Γ *and a morphism from a family of realized collections* A *to another family* B *is a realized morphism from* $\mathsf{pr}_{[\Gamma,A]}$ *to* $\mathsf{pr}_{[\Gamma,B]}$ *in the slice category* $\mathbf{Cont}/\Gamma$*. Composition and identities are inherited from those of* $\mathbf{Cont}/\Gamma$*.*

Lemma 4.16. *Let Γ be an object of* **Cont** *with $\ell(\Gamma) > 0$ and A, B be objects of* $\mathbf{Col}(\Gamma)$. *If* $\mathbf{n}$ *is a numeral for which*

$$\overline{x}_{|_{\ell(\Gamma)+1}} \sim_{[\Gamma,A]} \overline{y}_{|_{\ell(\Gamma)+1}} \vdash_{\widehat{ID_1}} \{\mathbf{n}\}(\overline{x}_{|_{\ell(\Gamma)+1}}) \sim_B \{\mathbf{n}\}(\overline{y}_{|_{\ell(\Gamma)+1}})$$

then

$$\gamma^{A,B}_{\mathbf{n},\Gamma} := [\,\pi_1^{\ell(\Gamma)+1}, ..., \pi_{\ell(\Gamma)}^{\ell(\Gamma)+1}, \mathbf{n}\,]_{\approx_{[\Gamma,A],[\Gamma,B]}}$$

is a well defined realized morphism from A to B in $\mathbf{Col}(\Gamma)$.

Conversely, for every $f : A \to B$ in $\mathbf{Col}(\Gamma)$ there exists a numeral $\mathbf{n}$ for which $f = \gamma^{A,B}_{\mathbf{n},\Gamma}$, and in this case we say that f is represented by $\mathbf{n}$.

If $[\mathbf{n}]_{\approx}$ is an arrow from A to B in $\mathbf{Col}([\,])$, then we will denote it also by $\gamma^{A,B}_{\mathbf{n},[\,]}$.

We will omit A, B and Γ in the notation $\gamma^{A,B}_{\mathbf{n},\Gamma}$, when they will be clear from the context.

Lemma 4.17. *Suppose $f = [\overline{\mathbf{k}}]_{\approx_{\Gamma',\Gamma}} : \Gamma' \to \Gamma$ in* **Cont**.

1. *If $A(\overline{x}_{|_{\ell(\Gamma)}})$ is an object of $\mathbf{Col}(\Gamma)$, then the conditions*

   (a) $x \,\varepsilon\, \mathbf{Col}_f(A(\overline{x}_{|_{\ell(\Gamma)}})) \equiv^{def} \overline{x}_{|_{\ell(\Gamma')}} \,\varepsilon\, \Gamma' \wedge x \,\varepsilon\, A(\{\overline{\mathbf{k}}\}(\overline{x}_{|_{\ell(\Gamma')}}))$

   (b) $x \sim_{\mathbf{Col}_f(A(\overline{x}_{|_{\ell(\Gamma)}}))} y \equiv^{def} \overline{x}_{|_{\ell(\Gamma')}} \,\varepsilon\, \Gamma' \wedge x \sim_{A(\{\overline{\mathbf{k}}\}(\overline{x}_{|_{\ell(\Gamma')}}))} y$

   determine an object $\mathbf{Col}_f(A(\overline{x}_{|_{\ell(\Gamma)}}))$ of $\mathbf{Col}(\Gamma')$.

2. *If $g = \gamma_{\mathbf{n}}$ is an arrow in $\mathbf{Col}(\Gamma)$ from A to B, then the numeral*

   $$\mathbf{n}' := \Lambda \overline{x}_{|_{\ell(\Gamma')+1}}.\{\mathbf{n}\}(\{\overline{\mathbf{k}}\}(\overline{x}_{|_{\ell(\Gamma')}}), x_{\ell(\Gamma')+1})$$

   determines an arrow $\mathbf{Col}_f(g) := \gamma_{\mathbf{n}'}$ in $\mathbf{Col}(\Gamma')$ from $\mathbf{Col}_f(A)$ to $\mathbf{Col}_f(B)$.

Moreover, $\mathbf{Col}_f(h \circ g) = \mathbf{Col}_f(h) \circ \mathbf{Col}_f(g)$ if $g : A \to B$ and $h : B \to C$ are arrows in $\mathbf{Col}(\Gamma)$ and $\mathbf{Col}_f(id_A) = id_{\mathbf{Col}_f(A)}$ if A is an object of $\mathbf{Col}(\Gamma)$, i. e. $\mathbf{Col}_f$ is a functor from $\mathbf{Col}(\Gamma)$ to $\mathbf{Col}(\Gamma')$.

Moreover we have the following property.

Lemma 4.18. *If $f : \Gamma' \to \Gamma''$ and $g : \Gamma'' \to \Gamma'''$ are arrows in* **Cont** *and Γ is an object of* **Cont**, *then*

1. $\mathbf{Col}_{g \circ f} = \mathbf{Col}_f \circ \mathbf{Col}_g$

2. $\mathbf{Col}_{id_\Gamma} = \mathbf{id}_{\mathbf{Col}_f(\Gamma)}$

Now we are ready to define the indexed category of families of realized collections as follows:

Definition 4.19. *Let* $\mathbf{Col} : \mathbf{Cont}^{op} \to \mathbf{Cat}$ *be the functor defined by the pair of assignment* $\Gamma \mapsto \mathbf{Col}(\Gamma)$, $f \mapsto \mathbf{Col}_f$.

$\mathbf{Col}_f$ *is called* the substitution functor along f.

In the following lemma we introduce the notation of pullback projections which will be used later to characterize the interpretation of the substitution of terms in types and the interpretation of the context operation of weakening:

Lemma 4.20. *If* Γ *and* Γ' *are objects of* $\mathbf{Cont}$, $f : \Gamma' \to \Gamma$ *in* $\mathbf{Cont}$ *and* A *is an object in* $\mathbf{Col}(\Gamma)$, *then* $\mathbf{Col}_f(A)$ *fits into a pullback in* $\mathbf{Cont}$ *as follows*

$$\begin{tikzcd}
{[\Gamma', \mathbf{Col}_f(A)]} \arrow[r, "{q(f,[\Gamma,A])}"] \arrow[d, "\mathsf{pr}"] & {[\Gamma, A]} \arrow[d, "\mathsf{pr}"] \\
\Gamma' \arrow[r, "f"] & \Gamma
\end{tikzcd}$$

where, if f *is represented by the list* $[\mathbf{k}_1, ..., \mathbf{k}_{\ell(\Gamma)}]$, *then* $q(f, [\Gamma, A])$ *is represented by the list*

$$[\, \Lambda\overline{x}_{|_{\ell(\Gamma')+1}}.\{\mathbf{k}_1\}(\overline{x}_{|_{\ell(\Gamma')}}), ..., \Lambda\overline{x}_{|_{\ell(\Gamma')+1}}.\{\mathbf{k}_{\ell(\Gamma)}\}(\overline{x}_{|_{\ell(\Gamma')}}), \pi^{\ell(\Gamma')+1}_{\ell(\Gamma')+1}\,]$$

Now, we are going to describe the categorical structure of each fibre $\mathbf{Col}(\Gamma)$ for a fixed context Γ.

Hence, *in all the following lemmas* Γ *is an object of* $\mathbf{Cont}$ *with* $\ell(\Gamma) = n$.

We start by showing that each fibre $\mathbf{Col}(\Gamma)$ is closed under finite products.

Lemma 4.21. *The object*

$$\mathbf{1}^\Gamma := (\, \{x|\, \overline{x}_{|_n} \,\varepsilon\, \Gamma \,\wedge\, x = 0\},\ \overline{x}_{|_n} \,\varepsilon\, \Gamma \,\wedge\, x = 0 \,\wedge\, x = y\,)$$

is a terminal object in $\mathbf{Col}(\Gamma)$, *i. e. for every* A *in* $\mathbf{Col}(\Gamma)$, *there exists a unique arrow* $!_{A,\mathbf{1}^\Gamma} : A \to \mathbf{1}^\Gamma$ *in* $\mathbf{Col}(\Gamma)$.

Lemma 4.22. *If* A *and* B *are objects of* $\mathbf{Col}(\Gamma)$, *then the object* $A \times^\Gamma B$ *defined by the following conditions:*

1. $x \,\varepsilon\, A \times^\Gamma B \equiv^{def} p_1(x) \,\varepsilon\, A \,\wedge\, p_2(x) \,\varepsilon\, B$
2. $x \sim_{A\times^\Gamma B} y \equiv^{def} p_1(x) \sim_A p_1(y) \,\wedge\, p_2(x) \sim_B p_2(y)$

with $\mathbf{p}_i^\Gamma := \Lambda\overline{x}_{|_{n+1}}.\{\mathbf{p}_i\}(x_{n+1})$ *for* $i = 1, 2$ *yields the following* binary product *diagram in* $\mathbf{Col}(\Gamma)$

$$\begin{tikzcd} A & A \times^\Gamma B \arrow[l, "\pi_1^{A,B} := \gamma_{\mathbf{p}_1^\Gamma}"'] \arrow[r, "\pi_2^{A,B} := \gamma_{\mathbf{p}_2^\Gamma}"'] & B \end{tikzcd}$$

i. e. for every $f : C \to A$ *and* $g : C \to B$ *in* $\mathbf{Col}(\Gamma)$*, there exists a unique arrow* $\langle f, g\rangle : C \to A \times^\Gamma B$ *in* $\mathbf{Col}(\Gamma)$ *for which the following diagram commutes*

$$\begin{tikzcd} A & A \times^\Gamma B \arrow[l, "\pi_1^{A,B}"'] \arrow[r, "\pi_2^{A,B}"] & B \\ & C \arrow[ul, "f"] \arrow[u, "\langle f,g\rangle"'] \arrow[ur, "g"'] & \end{tikzcd}$$

Now we are going to show how to form equalizers in $\mathbf{Col}(\Gamma)$.

Lemma 4.23. *If* A *is an object of* $\mathbf{Col}(\Gamma)$ *and* $f, g : \mathbf{1}^\Gamma \to A$ *are arrows in* $\mathbf{Col}(\Gamma)$ *represented by numerals* $\mathbf{n}_f$ *and* $\mathbf{n}_g$ *respectively, then* $\mathbf{Eq}^\Gamma(A, f, g)$ *given by the following conditions*

1. $x \,\varepsilon\, \mathbf{Eq}^\Gamma(A, f, g) \equiv^{def} \{\mathbf{n}_f\}(\overline{x}_{|_n}, 0) \sim_A \{\mathbf{n}_g\}(\overline{x}_{|_n}, 0)$
2. $x \sim_{\mathbf{Eq}^\Gamma(A,f,g)} y \equiv^{def} x \,\varepsilon\, \mathbf{Eq}^\Gamma(A, f, g) \wedge y \,\varepsilon\, \mathbf{Eq}^\Gamma(A, f, g)$

is a well defined object of $\mathbf{Col}(\Gamma)$*.*

Lemma 4.24. *Suppose* $f_1, f_2 : A \to B$ *in* $\mathbf{Col}(\Gamma)$ *and* $f_i = \gamma_{\mathbf{n}_i}$ *for* $i = 1, 2$*. If for* $i = 1, 2$ *we define* f_i' *to be* $\gamma_{\mathbf{n}'_i} : \mathbf{1}^{[\Gamma,A]} \to \mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(B)$ *in* $\mathbf{Col}([\Gamma, A])$ *with* $\mathbf{n}_i' := \Lambda\overline{x}_{|_{n+2}}.\{\mathbf{n}_i\}(\overline{x}_{|_{n+1}})$*, then*

$$E(f_1, f_2) := \Sigma^\Gamma(\, A,\, \mathbf{Eq}^{[\Gamma,A]}(\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(B), f_1', f_2')\,)$$

$$e(f_1, f_2) := \gamma_{\mathbf{p}_1^\Gamma} : E(f_1, f_2) \to A$$

define an equalizer for f_1 and f_2 in $\mathbf{Col}(\Gamma)$*, i. e. for every* $e' : E' \to A$ *for which* $f_1 \circ e' = f_2 \circ e'$*, there exists a unique arrow* $g : E' \to E(f_1, f_2)$ *in* $\mathbf{Col}(\Gamma)$ *for which the following diagram commutes.*

$$\begin{tikzcd} E(f_1, f_2) \arrow[r, "e(f_1,f_2)"] & A \\ & E' \arrow[ul, "g"] \arrow[u, "e'"'] \end{tikzcd}$$

Here we are going to show that each fibre $\mathbf{Col}(\Gamma)$ is closed under finite coproducts.

Lemma 4.25. *The object* $\mathbf{0}^\Gamma := (\,\{x|\perp\},\ \perp\,)$ *is an initial object of* $\mathbf{Col}(\Gamma)$*, i. e. for every* A *in* $\mathbf{Col}(\Gamma)$ *there exists a unique arrow* $!_{\mathbf{0}^\Gamma,A} : \mathbf{0}^\Gamma \to A$ *in* $\mathbf{Col}(\Gamma)$*.*

Lemma 4.26. *If* A *and* B *are objects of* $\mathbf{Col}(\Gamma)$*, then the object* $A +^\Gamma B$ *of* $\mathbf{Col}(\Gamma)$ *defined by the following conditions:*

1. $x\,\varepsilon\,A +^\Gamma B \equiv^{def} (\,p_1(x) = 0 \wedge p_2(x)\,\varepsilon\,A\,) \vee (\,p_1(x) = 1 \wedge p_2(x)\,\varepsilon\,B\,)$
2. $x \sim_{A+^\Gamma B} y \equiv^{def} p_1(x) = p_1(y) \wedge$

$$(\,(\,p_1(x) = 0 \wedge p_2(x) \sim_A p_2(y)\,) \vee (\,p_1(x) = 1 \wedge p_2(x) \sim_B p_2(y)\,)\,)$$

with $\mathbf{j}_1^\Gamma := \Lambda\overline{x}_{|n+1}.\{\mathbf{p}\}(0, x_{n+1})$ *and* $\mathbf{j}_2^\Gamma := \Lambda\overline{x}_{|n+1}.\{\mathbf{p}\}(1, x_{n+1})$ *yields the following* binary coproduct *diagram in* $\mathbf{Col}(\Gamma)$

$$A \xrightarrow{\mathsf{j}_1^{A,B} := \gamma_{\mathbf{j}_1^\Gamma}} A +^\Gamma B \xleftarrow{\mathsf{j}_2^{A,B} := \gamma_{\mathbf{j}_2^\Gamma}} B$$

i. e. for every object C *of* $\mathbf{Col}(\Gamma)$ *and every pair of arrows* $f : A \to C$ *and* $g : B \to C$*, there is a unique arrow* $\mathsf{case}(f, g) : A +^\Gamma B \to C$ *in* $\mathbf{Col}(\Gamma)$ *for which the following diagram commutes*

$$\begin{tikzcd}
A \arrow[r, "\mathsf{j}_1^{A,B}"] \arrow[dr, "f"'] & A +^\Gamma B \arrow[d, "\mathsf{case}(f,g)"] & B \arrow[l, "\mathsf{j}_2^{A,B}"'] \arrow[dl, "g"] \\
 & C &
\end{tikzcd}$$

Now we are going to show how $\mathbf{Col}(\Gamma)$ is closed under exponential objects, namely under function spaces:

Lemma 4.27. *If* A *and* B *are objects of* $\mathbf{Col}(\Gamma)$*, then the object* $A \Rightarrow^\Gamma B$ *defined by*

1. $x\,\varepsilon\,A \Rightarrow^\Gamma B \equiv^{def} \overline{x}_{|n}\,\varepsilon\,\Gamma \wedge \forall t \forall s\,(\,t \sim_A s \to \{x\}(t) \sim_B \{x\}(s)\,)$
2. $x \sim_{A\Rightarrow^\Gamma B} y \equiv^{def} x\,\varepsilon\,A \Rightarrow^\Gamma B \wedge y\,\varepsilon\,A \Rightarrow^\Gamma B \wedge \forall t\,(\,t\,\varepsilon\,A \to \{x\}(t) \sim_B \{y\}(t)\,)$

together with the arrow

$$\mathsf{ev}^{A,B} := \gamma_{\mathbf{ev}^\Gamma} : (A \Rightarrow^\Gamma B) \times^\Gamma A \to B$$

where $\mathbf{ev}^\Gamma$ *is* $\Lambda\overline{x}_{|n+1}.\{\{\mathbf{p}_1\}(x_{n+1})\}(\{\mathbf{p}_2\}(x_{n+1}))$ *defines an* exponential of A and B in $\mathbf{Col}(\Gamma)$ *i. e. for every object* C *of* $\mathbf{Col}(\Gamma)$ *and every arrow* $f : C \times^\Gamma A \to B$ *in*

$\mathbf{Col}(\Gamma)$*, there exists a unique arrow* $\mathsf{Cur}(f) : C \to A \Rightarrow^{\Gamma} B$ *for which the following diagram commutes in* $\mathbf{Col}(\Gamma)$[6]

$$\begin{tikzcd}
C \times^{\Gamma} A \arrow[r, "f"] \arrow[d, "\mathsf{Cur}(f)\times^{\Gamma} id_A"'] & B \\
(A \Rightarrow^{\Gamma} B) \times^{\Gamma} A \arrow[ur, "\mathsf{ev}^{A,B}"'] &
\end{tikzcd}$$

$\mathbf{Col}(\Gamma)$ has also list objects (see for instance [22] for a categorical definition)

Lemma 4.28. *If A is an object of $\mathbf{Col}(\Gamma)$, then the object $\mathbf{List}^{\Gamma}(A)$ defined by*

1. $x \,\varepsilon\, \mathbf{List}^{\Gamma}(A) \equiv^{def} \overline{x}_{|n} \,\varepsilon\, \Gamma \wedge \forall j\,(\, j < lh(x) \rightarrow (x)_j \,\varepsilon\, A\,)$
2. $x \sim_{\mathbf{List}^{\Gamma}(A)} y \equiv^{def} \overline{x}_{|n} \,\varepsilon\, \Gamma \wedge lh(x) = lh(y) \wedge \forall j\,(\, j < lh(x) \rightarrow (x)_j \sim_A (y)_j\,)$

together with the arrows

$$\epsilon^{A} := \gamma_{\Lambda \overline{x}_{|n+1}.0} : \mathbf{1}^{\Gamma} \to \mathbf{List}^{\Gamma}(A)$$

$$\mathsf{cons}^{A} := \gamma_{\mathbf{cnc}^{\Gamma}} : \mathbf{List}^{\Gamma}(A) \times^{\Gamma} A \to \mathbf{List}^{\Gamma}(A)$$

where $\mathbf{cnc}^{\Gamma}$ *is* $\Lambda \overline{x}_{|n+1}.\{\mathbf{cnc}\}(\{\mathbf{p}_1\}(x_{n+1}), \{\mathbf{p}_2\}(x_{n+1}))$*, defines a* list object on A in $\mathbf{Col}(\Gamma)$*, i. e. for every object B of $\mathbf{Col}(\Gamma)$ and every pair of arrows $f : \mathbf{1}^{\Gamma} \to B$ and $g : B \times^{\Gamma} A \to B$ in $\mathbf{Col}(\Gamma)$, there exists a unique arrow*

$$listrec(f, g) : \mathbf{List}^{\Gamma}(A) \to B$$

for which the following diagram commutes in $\mathbf{Col}(\Gamma)$

$$\begin{tikzcd}
\mathbf{1}^{\Gamma} \arrow[r, "\epsilon^{A}"] \arrow[dr, "f"'] & \mathbf{List}^{\Gamma}(A) \arrow[d, "listrec(f,g)"] & \mathbf{List}^{\Gamma}(A) \times^{\Gamma} A \arrow[l, "\mathsf{cons}^{A}"'] \arrow[d, "listrec(f,g)\times^{\Gamma} id_A"] \\
& B & B \times^{\Gamma} A \arrow[l, "g"]
\end{tikzcd}$$

Theorem 4.29. *For every Γ in $\mathbf{Cont}$, $\mathbf{Col}(\Gamma)$ is a finitely complete cartesian closed category with finite coproducts and list objects and for every morphism f in $\mathbf{Cont}$ the functors $\mathbf{Col}_f$ preserve this structure.*

[6]For $f : A \to C$ and $g : B \to D$ in $\mathbf{Col}(\Gamma)$, we use the notation $f \times^{\Gamma} g$ for the arrow $\langle f \circ \pi_1^{A,B}, g \circ \pi_2^{A,B}\rangle : A \times^{\Gamma} B \to C \times^{\Gamma} D$.

Proof. This is a consequence of the previous lemmas (see [20]) and it is an immediate verification to see that all these structures are preserved by the functors $\mathbf{Col}_f$. $\square$

Remark 4.30. *The object* $\mathbf{N}^\Gamma$ *of* $\mathbf{Col}(\Gamma)$ *defined by the following:*

1. $x \,\varepsilon\, \mathbf{N}^\Gamma \equiv^{def} \overline{x}_{|_{\ell(\Gamma)}} \,\varepsilon\, \Gamma \wedge x = x$
2. $x \sim_{\mathbf{N}^\Gamma} y \equiv^{def} \overline{x}_{|_{\ell(\Gamma)}} \,\varepsilon\, \Gamma \wedge x = y$

together with the arrows

$$\mathsf{z}^\Gamma := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)+1}}.0} : \mathbf{1}^\Gamma \to \mathbf{N}^\Gamma \qquad \mathsf{s}^\Gamma := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)+1}}.\{\mathbf{s}\}(x_{\ell(\Gamma)+1})} : \mathbf{N}^\Gamma \to \mathbf{N}^\Gamma.$$

defines a natural numbers object in $\mathbf{Col}(\Gamma)$*, i. e. for every A in* $\mathbf{Col}(\Gamma)$ *and for every pair of arrows $f : \mathbf{1}^\Gamma \to A$ and $g : A \to A$ in* $\mathbf{Col}(\Gamma)$*, there exists a unique arrow $rec(f, g) : \mathbf{N}^\Gamma \to A$ for which the following diagram commutes.*

$$\begin{tikzcd}
\mathbf{1}^\Gamma \arrow[r, "\mathsf{z}^\Gamma"] \arrow[dr, "f"'] & \mathbf{N}^\Gamma \arrow[r, "\mathsf{s}^\Gamma"] \arrow[d, "rec(f,g)"] & \mathbf{N}^\Gamma \arrow[d, "rec(f,g)"] \\
& A \arrow[r, "g"'] & A
\end{tikzcd}$$

It is immediate to see that this natural numbers object is preserved by the substitution functors $\mathbf{Col}_f$*. A natural numbers object can be defined also as* $\mathbf{List}^\Gamma(\mathbf{1}^\Gamma)$*, but it is convenient to consider the representation* $\mathbf{N}^\Gamma$ *to simplify the realizability interpretation of* $\mathbf{mTT}^a$*.*

Corollary 4.31. **Cont** *is a finitely complete cartesian closed category with finite coproducts and list objects.*

Proof. This is an immediate consequence of theorem 4.29 and 4.13 as $\mathcal{C}$ is clearly isomorphic to $\mathbf{Col}([\,])$. $\square$

Definition 4.32. *If $f : [\Gamma, \mathbf{1}^\Gamma] \to [\Gamma, A]$ is an arrow in* **Cont***, then we define $\widetilde{f} : \Gamma \to [\Gamma, A]$ as $f \circ j$ where $j : \Gamma \to [\Gamma, \mathbf{1}^\Gamma]$ is the isomorphism in* **Cont** *defined by the list*

$$[\,\pi_1^{\ell(\Gamma)}, ..., \pi_{\ell(\Gamma)}^{\ell(\Gamma)}, \Lambda \overline{x}_{|_{\ell(\Gamma)}}.0\,]$$

Now, we are going to show that, for any realized collection A in $\mathbf{Col}(\Gamma)$ there are left adjoints to substitution functors of the kind $\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}$ which will be used to interpret the operation of weakening the context Γ to $[\Gamma, A]$. These left adjoints will be used to interpret the strong indexed sum collections of $\mathbf{mTT}^a$.

Lemma 4.33. *Suppose* Γ *is an object in* $\mathbf{Cont}$ *and* A *is an object in* $\mathbf{Col}(\Gamma)$. *Then the functor sending each* B *in* $\mathbf{Col}([\Gamma, A])$ *to* $\Sigma^\Gamma(A, B)$ *and each arrow* $f := \gamma_{\mathbf{n}} : B \to C$ *in* $\mathbf{Col}([\Gamma, A])$ *to the arrow* $\Sigma^\Gamma(A, f)$ *from* $\Sigma^\Gamma(A, B)$ *to* $\Sigma^\Gamma(A, C)$ *in* $\mathbf{Col}([\Gamma])$ *represented by* $\Lambda\overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(\{\mathbf{p}_1\}(x), \{\mathbf{n}\}(\overline{x}_{|_{\ell(\Gamma)}}, \{\mathbf{p}_1\}(x), \{\mathbf{p}_2\}(x)))$, *in the sense of lemma 4.16, is left adjoint to the functor* $\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}$, *i. e. there is a bijection (see [20])*

$$Hom_{\mathbf{Col}(\Gamma)}(\Sigma^\Gamma(A, B), D) \cong Hom_{\mathbf{Col}([\Gamma,A])}(B, \mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(D))$$

natural in every B *in* $\mathbf{Col}([\Gamma, A])$ *and* D *in* $\mathbf{Col}(\Gamma)$.

We also give the following lemma which will be useful for the interpretation.

Lemma 4.34. *For every* Γ *in* $\mathbf{Cont}$ *and for every* A *in* $\mathbf{Col}(\Gamma)$ *and* B *in* $\mathbf{Col}([\Gamma, A])$, *the object* $\Sigma^\Gamma(A, B)$ *satisfies the following properties. If* $\mathsf{p}_1^\Sigma := \gamma_{\mathbf{p}_1^\Gamma} : \Sigma^\Gamma(A, B) \to A$ *(see lemma 4.22) in* $\mathbf{Col}(\Gamma)$, *for every* $f : \mathbf{1} \to A$ *in* $\mathbf{Col}(\Gamma)$ *and* $g : \mathbf{1} \to \mathbf{Col}_{\widetilde{f}}(B)$ *in* $\mathbf{Col}(\Gamma)$, *there is a unique arrow* $\langle f, g\rangle_\Sigma : \mathbf{1} \to \Sigma^\Gamma(A, B)$ *in* $\mathbf{Col}(\Gamma)$ *for which the following diagrams commute (the first in* $\mathbf{Col}(\Gamma)$, *the second in* $\mathbf{Cont}$*)*

$$\begin{tikzcd} \mathbf{1} \arrow[r, "\langle f,g\rangle_\Sigma"] \arrow[d, "f"'] & \Sigma^\Gamma(A,B) \arrow[dl, "\mathsf{p}_1^\Sigma"] \\ A & \end{tikzcd} \qquad \begin{tikzcd} {[\Gamma, \mathbf{1}]} \arrow[r, "\langle f,g\rangle_\Sigma"] \arrow[d, "g"] & {[\Gamma, \Sigma^\Gamma(A,B)]} \arrow[d, "\simeq"] \\ {[\Gamma, \mathbf{Col}_{\widetilde{f}}(B)]} \arrow[r, "q(\widetilde{f}, {[\Gamma,A,B]})"] & {[\Gamma, A, B]} \end{tikzcd}$$

where $\simeq$ *is the isomorphism from* $[\Gamma, \Sigma^\Gamma(A, B)]$ *to* $[\Gamma, A, B]$ *defined in lemma 4.12.*

Conversely, for every $h : \mathbf{1} \to \Sigma^\Gamma(A, B)$ *in* $\mathbf{Col}(\Gamma)$, *there is a unique arrow*

$$\mathsf{p}_2^\Sigma(h) : \mathbf{1} \to \mathbf{Col}_{\widetilde{\mathsf{p}_1^\Sigma \circ h}}(B)$$

in $\mathbf{Col}(\Gamma)$ *for which the following diagram commutes in* $\mathbf{Cont}$

$$\begin{tikzcd} {[\Gamma, \mathbf{1}]} \arrow[r, "h"] \arrow[d, "\mathsf{p}_2^\Sigma(h)"] & {[\Gamma, \Sigma^\Gamma(A,B)]} \arrow[d, "\simeq"] \\ {[\Gamma, \mathbf{Col}_{\widetilde{\mathsf{p}_1^\Sigma \circ h}}(B)]} \arrow[r, "q(\widetilde{\mathsf{p}_1^\Sigma \circ h}, {[\Gamma,A,B]})"] & {[\Gamma, A, B]} \end{tikzcd}$$

where $\widetilde{f}$ *and* $\widetilde{\mathsf{p}_1^\Sigma \circ h}$ *are as in definition 4.32.*

Now, we are going to show that, for any realized collection A in $\mathbf{Col}(\Gamma)$ there are right adjoints to substitution functors of the kind $\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}$. These right adjoints will be used to interpret the dependent product sets of $\mathbf{mTT}^a$.

Definition 4.35. *Let* Γ *be an object of* $\mathbf{Cont}$ *with* $\ell(\Gamma) = n$, $A(\overline{x}_{|n})$ *an object of* $\mathbf{Col}(\Gamma)$ *and* $B(\overline{x}_{|n+1})$ *an object of* $\mathbf{Col}([\Gamma, A(\overline{x}_{|n})])$. *We define* $\Pi^\Gamma(A(\overline{x}_{|n}), B(\overline{x}_{|n+1}))$ *as follows:*

1. $x \,\varepsilon\, \Pi^\Gamma(A(\overline{x}_{|n}), B(\overline{x}_{|n+1})) \equiv^{def}$

$$\overline{x}_{|n} \,\varepsilon\, \Gamma \wedge \forall t \forall s\, (t \sim_{A(\overline{x}_{|n})} s \rightarrow \{x\}(t) \sim_{B(\overline{x}_{|n},t)} \{x\}(s));$$

2. $x \sim_{\Pi^\Gamma(A(\overline{x}_{|n}),B(\overline{x}_{|n+1}))} y \equiv^{def} x \,\varepsilon\, \Pi^\Gamma(A(\overline{x}_{|n}), B(\overline{x}_{|n+1})) \wedge$

$$y \,\varepsilon\, \Pi^\Gamma(A(\overline{x}_{|n}), B(\overline{x}_{|n+1})) \wedge \forall t\, (t \,\varepsilon\, A(\overline{x}_{|n}) \rightarrow \{x\}(t) \sim_{B(\overline{x}_{|n},t)} \{y\}(t)).$$

Lemma 4.36. *Suppose* Γ *is an object in* $\mathbf{Cont}$ *and* A *is an object in* $\mathbf{Col}(\Gamma)$. *Then the functor sending each object* B *in* $\mathbf{Col}([\Gamma, A])$ *to* $\Pi^\Gamma(A, B)$ *and each arrow* $f := \gamma_{\mathbf{n}} : B \rightarrow C$ *in* $\mathbf{Col}([\Gamma, A])$ *to the arrow* $\Pi^\Gamma(A, f)$ *from* $\Pi^\Gamma(A, B)$ *to* $\Pi^\Gamma(A, C)$ *in* $\mathbf{Col}([\Gamma])$ *represented by* $\Lambda \overline{x}_{|\ell(\Gamma)}.\Lambda x.\Lambda y.\{\mathbf{n}\}(\overline{x}_{|\ell(\Gamma)}, y, \{x\}(y))$, *in the sense of lemma 4.16, is right adjoint to the functor* $\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}$, *i. e. there is a bijection (see [20])*

$$Hom_{\mathbf{Col}(\Gamma)}(D, \Pi^\Gamma(A, B)) \cong Hom_{\mathbf{Col}([\Gamma,A])}(\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(D), B)$$

natural in every B *in* $\mathbf{Col}([\Gamma, A])$ *and* D *in* $\mathbf{Col}(\Gamma)$.

Corollary 4.37. *For every* Γ *in* $\mathbf{Cont}$, *for every* A *and* C *in* $\mathbf{Col}(\Gamma)$ *and for every* B *in* $\mathbf{Col}([\Gamma, A])$, *the object* $\Pi^\Gamma(A, B)$ *satisfies the following universal property: there is an arrow* ev^Γ_Π *from* $\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(\Pi^\Gamma(A, B))$ *to* B *in* $\mathbf{Col}([\Gamma, A])$ *such that for every* $f : \mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(C) \rightarrow B$ *in* $\mathbf{Col}([\Gamma, A])$, *there exists a unique arrow*

$$\mathsf{Cur}_\Pi(f) : C \rightarrow \Pi^\Gamma(A, B)$$

in $\mathbf{Col}(\Gamma)$ *for which the following diagram commutes in* $\mathbf{Col}([\Gamma, A])$:

$$\begin{tikzcd}
\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(C) \arrow[r, "f"] \arrow[d, "\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(\mathsf{Cur}_\Pi(f))"'] & B \\
\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(\Pi^\Gamma(A, B)) \arrow[ur, "\mathsf{ev}^\Gamma_\Pi"'] &
\end{tikzcd}$$

Observe that the substitution functor $\mathbf{Col}_f$ along any morphism f of **Cont** preserves left and right adjoints described above as follows:

Lemma 4.38. *Suppose $f : \Gamma' \to \Gamma$ in* **Cont**, *A is an object of* $\mathbf{Col}(\Gamma)$, *$f' := q(f, [\Gamma, A])$ and B is an object of* $\mathbf{Col}([\Gamma, A])$. *Then*

1. $\mathbf{Col}_f(\Sigma^\Gamma(A, B)) = \Sigma^{\Gamma'}(\mathbf{Col}_f(A), \mathbf{Col}_{f'}(B))$
2. $\mathbf{Col}_f(\Pi^\Gamma(A, B)) = \Pi^{\Gamma'}(\mathbf{Col}_f(A), \mathbf{Col}_{f'}(B))$

Note here that left adjoints and right adjoints to substitution functors of the kind $\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}$ provide respectively binary products and exponentials as follows:

Lemma 4.39. *If Γ is an object of* **Cont** *and A, B are objects of* $\mathbf{Col}(\Gamma)$, *then*

$$\Sigma^\Gamma(A, \mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(B))) = A \times^\Gamma B \qquad \Pi^\Gamma(A, \mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}(B))) = A \Rightarrow^\Gamma B$$

The following lemma will be useful in the interpretation of $\mathbf{mTT}^a$-eliminators for disjoint sums, natural numbers and lists.

Lemma 4.40. *If Γ is an object of* **Cont**, *$A_1, ..., A_n, B$ are objects of* $\mathbf{Col}(\Gamma)$ *and*

1. $\widetilde{A_1} := A_1$
2. $\widetilde{A_{i+1}} := \mathbf{Col}_{\mathsf{pr}^{(i)}_{[\Gamma,\widetilde{A_1},...,\widetilde{A_i}]}}(A_{i+1})$ *for* $i = 1, .., n-1$
3. $\widetilde{B} := \mathbf{Col}_{\mathsf{pr}^{(n)}_{[\Gamma,\widetilde{A_1},...,\widetilde{A_n}]}}(B)$

and $f := \gamma_{\mathbf{n}} : \mathbf{1} \to \widetilde{B}$ *in* $\mathbf{Col}([\Gamma, \widetilde{A_1}, ..., \widetilde{A_n}])$, *then*

$$f^\Gamma_\natural := \gamma_{\mathbf{n}'} : ((A_1 \times^\Gamma A_2) \times .... \times^\Gamma A_n) \to B$$

where $\mathbf{n}'$ is defined as

$$\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{n}\}(\overline{x}_{|_{\ell(\Gamma)}}, \{\mathbf{p}^n_1\}(x), ..., \{\mathbf{p}^n_n\}(x), 0)$$

is a well defined morphism in $\mathbf{Col}(\Gamma)$.

The left and right adjoints to the substitution functors of the kind $\mathbf{Col}_{\mathsf{pr}_{[\Gamma,A]}}$ are enough to provide left and right adjoints to substitution functors along any arrow in **Cont** (see for example [26] and loc.cit. for a proof):

Corollary 4.41. *For any arrow f in* **Cont**, *the substitution functor* $\mathbf{Col}_f$ *enjoys left and right adjoints satisfying Beck-Chevalley conditions.*

Moreover, the category **Cont** *is locally cartesian closed (see for instance [36] for a definition).*

4.4 Families of realized propositions as an indexed category

Here we are going to define an indexed category of realized propositions equipped with the structure of a first-order Lawvere hyperdoctrine

$$\mathbf{Prop} : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

on the category of realized contexts. This will be used to interpret generic propositions of $\mathbf{mTT}^a$.

We start by giving a lemma characterizing a proof-irrelevant dependent realized collection, namely a realized collection with at most one element:

Lemma 4.42. *Let Γ be an object of $\mathbf{Cont}$ and let P be an object of $\mathbf{Col}(\Gamma)$. Then the following conditions are equivalent:*

1. *for every object A in $\mathbf{Col}(\Gamma)$, if $f, g : A \to P$ are arrows in $\mathbf{Col}(\Gamma)$, then $f = g$;*
2. $\pi_1^{P,P} = \pi_2^{P,P} : P \times^\Gamma P \to P$*;*
3. *P is* proof-irrelevant*, i.e.* $x \,\varepsilon\, P \wedge y \,\varepsilon\, P \vdash_{\widehat{ID_1}} x \sim_P y$*.*

Now we define the notion of *a family of realized propositions* as a proof-irrelevant dependent realized collection:

Definition 4.43. *Let Γ be an object of $\mathbf{Cont}$. A* family of realized propositions *on Γ (or a* realized proposition depending on Γ*) is a proof-irrelevant object of $\mathbf{Col}(\Gamma)$ as in lemma 4.42.*

Definition 4.44. *Let $\mathbf{Prop}(\Gamma)$ be the full subcategory of $\mathbf{Col}(\Gamma)$ whose objects are families of realized propositions on Γ.*

Observe that $\mathbf{Prop}(\Gamma)$ is a preorder, as a consequence of point 1. in lemma 4.42. Hence we put:

Definition 4.45. *If Γ is an object of $\mathbf{Cont}$ and P and Q are in $\mathbf{Prop}(\Gamma)$, we write*

$$P \sqsubseteq^\Gamma Q$$

if there is an arrow in $\mathbf{Prop}(\Gamma)$ from P to Q. Moreover, if the existing arrow is called f then we may write

$$f : P \sqsubseteq^\Gamma Q$$

Moreover, observe also that, contrary to what happens in **mTT**, in our indexed category of dependent realized collections it is possible to transform any dependent realized collection into a dependent realized proposition by quotienting it under the trivial relation:

Definition 4.46. *If* Γ *is an object of* $\mathbf{Cont}$ *and* A *is an object of* $\mathbf{Col}(\Gamma)$*, then the* proof-irrelevant quotient $\mathbf{Pir}(A)$ of A *is the object of* $\mathbf{Prop}(\Gamma)$ *defined by the following conditions:*

1. $x \,\varepsilon\, \mathbf{Pir}(A) \equiv^{def} x \,\varepsilon\, A$
2. $x \sim_{\mathbf{Pir}(A)} y \equiv^{def} x \,\varepsilon\, A \wedge y \,\varepsilon\, A$

Actually, the above operation defines a reflector (see [20] for a definition) from realized collections to propositions:

Lemma 4.47. *For every object* Γ *of* $\mathbf{Cont}$

$$\mathbf{Pir} : \mathbf{Col}(\Gamma) \longrightarrow \mathbf{Prop}(\Gamma)$$

defined as $\mathbf{Pir}(A)$ *for any object* A *of* $\mathbf{Col}(\Gamma)$ *and as* $\gamma_{\mathbf{n}} : \mathbf{Pir}(A) \to \mathbf{Pir}(B)$ *for every* $f := \gamma_{\mathbf{n}} : A \to B$ *in* $\mathbf{Col}(\Gamma)$*, is a reflector of the embedding functor of* $\mathbf{Prop}(\Gamma)$ *into* $\mathbf{Col}(\Gamma)$*. This means that there is a bijection*

$$Hom_{\mathbf{Prop}(\Gamma)}(\mathbf{Pir}(A), P) \cong Hom_{\mathbf{Col}(\Gamma)}(A, P)$$

natural in every object A *in* $\mathbf{Col}(\Gamma)$ *and object* P *in* $\mathbf{Prop}(\Gamma)$*.*

As a consequence, we get that each category of dependent propositions is an Heyting prealgebra:

Corollary 4.48. $\mathbf{Prop}(\Gamma)$ *is an Heyting prealgebra, i. e. it is a preorder with all binary infima and suprema, bottom and top elements and all Heyting implications i. e. it is a cartesian closed preorder category with finite coproducts.*

Proof. In order to show that $\mathbf{Prop}(\Gamma)$ is an Heyting prealgebra it is sufficient to show that it has binary infima and suprema, a bottom element, a top element and Heyting implications. A bottom element is given by $\bot^{\Gamma} := \mathbf{0}^{\Gamma}$, a top element is given by $\top^{\Gamma} := \mathbf{1}^{\Gamma}$, a binary supremum, a binary infimum and a Heyting implication for P and Q in $\mathbf{Prop}(\Gamma)$ are $P \sqcap^{\Gamma} Q := P \times^{\Gamma} Q$, $P \sqcup^{\Gamma} Q := \mathbf{Pir}(P +^{\Gamma} Q)$ and $P \to^{\Gamma} Q := P \Rightarrow^{\Gamma} Q$ respectively. $\square$

Observe that a substitution functor $\mathbf{Prop}_f$ along any arrow f in $\mathbf{Cont}$ is inherited from that of $\mathbf{Col}$:

Lemma 4.49. *Suppose $f : \Gamma \to \Gamma'$ in* **Cont** *and suppose P is in* $\mathbf{Prop}(\Gamma')$*, then* $\mathbf{Col}_f(P)$ *is in* $\mathbf{Prop}(\Gamma)$*. Moreover,* $\mathbf{Col}_f|_{\mathbf{Prop}(\Gamma')}$ *is a morphism of Heyting prealgebras, i. e. it preserves $\bot$, $\top$, $\sqcap$, $\sqcup$ and $\to$.*

Hence we are ready to define **Prop** as a *sub-indexed category* of **Col**:

Definition 4.50. *We call* $\mathbf{Prop} : \mathbf{Cont}^{op} \to \mathbf{Cat}$ *the indexed category defined by the assignments $A \mapsto \mathbf{Prop}(A)$ and $f \mapsto \mathbf{Prop}_f := \mathbf{Col}_f|_{\mathbf{Prop}(\mathsf{cod}(f))}$ where $\mathsf{cod}(f)$ denotes the codomain of f.*

Now we describe left and right adjoints to substitution functors which are necessary to interpret existential and universal quantifiers of $\mathbf{mTT}^a$ respectively:

Definition 4.51. *Suppose Γ is an object of* **Cont***, A is an object of* $\mathbf{Col}(\Gamma)$ *and P is an object of* $\mathbf{Prop}([\Gamma, A])$*, then*

1. $\exists^\Gamma(A, P) := \mathbf{Pir}(\Sigma^\Gamma(A, P))$
2. $\forall^\Gamma(A, P) := \mathbf{Pir}(\Pi^\Gamma(A, P))$

Observe that in $\mathbf{Prop}(\Gamma)$ there are also the propositional equalities of $\mathbf{Col}(\Gamma)$ and these are preserved by substitution functors:

Lemma 4.52. *If Γ is an object of* **Cont** *and $f, g : \mathbf{1}^\Gamma \to A$ in* $\mathbf{Col}(\Gamma)$*, then* $\mathbf{Eq}^\Gamma(A, f, g)$ *is an object of* $\mathbf{Prop}(\Gamma)$*. Morerover, if $\top^\Gamma \sqsubseteq^\Gamma \mathbf{Eq}^\Gamma(A, f, g)$, then f is equal to g in* $\mathbf{Col}(\Gamma)$*.*

Lemma 4.53. *Suppose $f : \Gamma' \to \Gamma$ is an arrow of* **Cont***, A is an object of* $\mathbf{Col}(\Gamma)$ *and $g, g' : \mathbf{1}^\Gamma \to A$ in* $\mathbf{Col}(\Gamma)$*. Then*

$$\mathbf{Col}_f(\mathbf{Eq}^\Gamma(A, g, g')) = \mathbf{Eq}^{\Gamma'}(\mathbf{Col}_f(A), \mathbf{Col}_f(g), \mathbf{Col}_f(g'))$$

Finally, observe that there exist left and right adjoints to substitution functors and that Beck-Chevalley conditions hold for them:

Lemma 4.54. *For every $f : \Gamma \to \Gamma'$ in* **Cont***,* $\mathbf{Prop}_f : \mathbf{Prop}(\Gamma') \to \mathbf{Prop}(\Gamma)$ *has a left and right adjoint* $\exists_f : \mathbf{Prop}(\Gamma) \to \mathbf{Prop}(\Gamma')$ *and* $\forall_f : \mathbf{Prop}(\Gamma) \to \mathbf{Prop}(\Gamma')$ *respectively, i. e. $\exists_f$ and $\forall_f$ are preorder morphims for which for every* $P \in \mathbf{Prop}(\Gamma')$ *and* $Q \in \mathbf{Prop}(\Gamma)$*:*

1. $Q \sqsubseteq^\Gamma \mathbf{Prop}_f(P)$ *if and only if* $\exists_f(Q) \sqsubseteq^{\Gamma'} P$*,*
2. $\mathbf{Prop}_f(P) \sqsubseteq^\Gamma Q$ *if and only if* $P \sqsubseteq^{\Gamma'} \forall_f(Q)$*.*

Moreover these adjoints satisfy the Beck-Chevalley condition: for every pullback square in **Cont**

$$\begin{tikzcd} \Gamma' \arrow[r, "f'"] \arrow[d, "g'"'] & \Delta' \arrow[d, "g"] \\ \Gamma \arrow[r, "f"] & \Delta \end{tikzcd}$$

and for every P in $\mathbf{Prop}(\Delta')$ *the following conditions hold*[7]*:*

1. $\exists_{g'}(\mathbf{Prop}_{f'}(P)) \sqsubseteq^{\Gamma} \mathbf{Prop}_f(\exists_g(P))$ *and* $\mathbf{Prop}_f(\exists_g(P)) \sqsubseteq^{\Gamma} \exists_{g'}(\mathbf{Prop}_{f'}(P))$*;*

2. $\forall_{g'}(\mathbf{Prop}_{f'}(P)) \sqsubseteq^{\Gamma} \mathbf{Prop}_f(\forall_g(P))$ *and* $\mathbf{Prop}_f(\forall_g(P)) \sqsubseteq^{\Gamma} \forall_{g'}(\mathbf{Prop}_{f'}(P))$*;*

Proof. Note that if A is an object of $\mathbf{Col}(\Gamma)$ and P is an object of $\mathbf{Prop}([\Gamma, A])$, then we can define $\exists_{\mathsf{pr}_{[\Gamma,A]}}(P)$ and $\forall_{\mathsf{pr}_{[\Gamma,A]}}(P)$ as the objects $\exists^{\Gamma}(A, P)$ and $\forall^{\Gamma}(A, P)$ defined in 4.51 respectively. $\square$

From lemmas 4.47, 4.49 and 4.54 we conclude

Corollary 4.55. **Prop** *is a hyperdoctrine in the sense of [37] and its posetal reflection is a first-order hyperdoctrine in the sense of [32].*

4.5 Realized sets and small realized propositions

Here we are going to define a notion of realized set and of small realized proposition in order to define a sub-indexed category **Set** of **Col**

$$\mathbf{Set} : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

and a sub-indexed category of **Prop**

$$\mathbf{Prop}_s : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

which will be used to interpret $\mathbf{mTT}^a$-sets and $\mathbf{mTT}^a$-small propositions respectively. In order to interpret the $\mathbf{mTT}^a$-collection of small propositions and that of sets of section 2.3 following the interpretation in [24], both indexed categories will need to enjoy a classifier: the fibres of **Set** will need to be classified by an object US of $\mathbf{Col}([\,])$ via a natural bijection in Γ

$$\mathbf{Set}(\Gamma) \simeq \mathbf{Cont}(\Gamma, \mathsf{US})$$

[7]It is sufficient that one of the two conditions holds as the other follows by adjunction.

and the fibres of $\mathbf{Prop}_s$ will need to be classified by an object USP of $\mathbf{Col}([\,])$ via a natural bijection in Γ

$$\mathbf{Prop}_s(\Gamma) \simeq \mathbf{Cont}(\Gamma, \mathsf{USP})$$

Actually, we will define both indexed categories by using their classifiers: a realized set depending on Γ will be defined as the dependent realized collection made of elements of a code in US over Γ and, analogously, a small realized proposition will be defined as the proof-irrelevant dependent realized collection of elements of a code in USP over Γ. In turn the objects US and USP will be defined as those used in [24] to interpret the collection of sets and the collection of small propositions respectively. Both objects are realized collections according to the terminology used here.

We describe now the construction of US and USP which will make use of fixpoint formulas of $\widehat{ID_1}$ as in [24]. To this purpose, we start by recalling the definition of Kleene realizability for Heyting Arithmetic since it will be used to define the notion of element both of a set and of a proposition.

Definition 4.56. *For every formula φ of HA the formula $x \Vdash_k \varphi$ (x realizes φ) is defined according to the following clauses by external induction on the formation of formulas (x is a variable which is not free in φ).*

1. $x \Vdash_k t = s$ *is* $t = s$
2. $x \Vdash_k (\varphi \wedge \varphi')$ *is* $p_1(x) \Vdash_k \varphi \wedge p_2(x) \Vdash_k \varphi'$
3. $x \Vdash_k (\varphi \vee \varphi')$ *is* $(p_1(x) = 0 \wedge p_2(x) \Vdash_k \varphi) \vee (p_1(x) = 1 \wedge p_2(x) \Vdash_k \varphi')$
4. $x \Vdash_k (\varphi \to \varphi')$ *is* $\forall t\,(t \Vdash_k \varphi \to \{x\}(t) \Vdash_k \varphi')$
5. $x \Vdash_k \forall y\, \varphi$ *is* $\forall y\,(\{x\}(y) \Vdash_k \varphi)$
6. $x \Vdash_k \exists y\, \varphi$ *is* $p_2(x) \Vdash_k \varphi[p_1(x)/y]$

Then, we define the following *formulas in $\widehat{ID_1}$ as fixpoints*:

1. $Set(x)$ intended to state that x *is a code for a set of* $\mathbf{mTT}^a$;
2. $x\,\varepsilon\,y$ intended to state that x *is an element of the set of* $\mathbf{mTT}^a$ *coded by* y;
3. $x \not\varepsilon\, y$ intended to state that x *is not an element of the set of* $\mathbf{mTT}^a$ *coded by* y;
4. $x \equiv_z y$ intended to state that x *and* y *are equal elements in the set of* $\mathbf{mTT}^a$ *coded by* z;
5. $x \not\equiv_z y$ intended to state that x *and* y *are not equal elements of the set of* $\mathbf{mTT}^a$ *coded by* z.

We will use such formulas to encode with natural numbers a realizability interpretation of $\mathbf{mTT}^a$-sets in $\widehat{ID_1}$: we use natural numbers to represent both realizers and (codes for) sets and we introduce a membership relation $x \,\overline{\varepsilon}\, z$ between natural numbers (which extends the notion of Kleene realizability) and an equivalence relation $x \equiv_z y$ between numbers (realizers) of a (code of a) set z, which will represent the equality between its elements. These clauses are similar to those presented in Beeson's book ([8]) for the first-order fragment of Martin-Löf type theory with one universe, except that here we need to deal with $\mathbf{mTT}^a$-sets which include an extra notion of proposition, that of small proposition, defined primitively. As in Beeson's book we define also the formal negations $x \not\overline{\varepsilon}\, z$ and $x \not\equiv_z y$ in order to give admissible clauses defining the properties of our new formulas. Then, we will encode the set constructors N_0, N_1, N, Σ, Π, $+$, List, $\bot$, $\wedge$, $\vee$, $\rightarrow$, $\exists$, $\forall$, Eq. In order to mimic the dependency, we define a family of sets on a given set as (a code for) a recursive function defined on the elements of a (code for a) set and producing codes for sets as outputs provided that some coherence requirements are fulfilled. Formally, we introduce the formula $Fam(y, x)$ (*y is a family of sets on the set x*) in order to capture this idea:

$$Fam(y,x) \equiv^{def} Set(x) \wedge \forall t\,(t \not\overline{\varepsilon}\, x \vee Set(\{y\}(t))) \wedge \forall t \forall s\,(t \not\equiv_x s \vee \{y\}(t) =_{ext\#} \{y\}(s))$$

where $x =_{ext\#} y$ is defined as

$$\forall t\,((t \,\overline{\varepsilon}\, x \ \vee\ t \not\overline{\varepsilon}\, y) \ \wedge\ (t \,\overline{\varepsilon}\, y \ \vee\ t \not\overline{\varepsilon}\, x)) \wedge \forall t \forall s\,((t \equiv_x s \ \vee\ t \not\equiv_y s) \ \wedge\ (t \equiv_y s \ \vee\ t \not\equiv_x s))$$

We then declare that y *is a family of small propositions* with the abbreviation $Fam_p(y, x)$ defined formally as

$$Fam(y,x) \wedge \forall t\,(t \not\overline{\varepsilon}\ x \ \vee\ p_1(\{y\}(t)) > 5)$$

where the last condition means that $\{y\}(t)$ is a small proposition (see later explanations).

For every constructor κ the clauses for the definitions of the fixpoint formulas are described by using extra new formulas as follows

1. $Set(\kappa^{\#})$ if $Cond(\kappa)$
2. $x \,\overline{\varepsilon}\, \kappa^{\#}$ if $Cond(\kappa) \wedge P^{\kappa}_{\overline{\varepsilon}}(x)$
3. $x \not\overline{\varepsilon}\, \kappa^{\#}$ if $Cond(\kappa) \wedge \overline{P^{\kappa}_{\overline{\varepsilon}}(x)}$
4. $x \equiv_{\kappa^{\#}} y$ if $Cond(\kappa) \wedge P^{\kappa}_{\overline{\varepsilon}}(x) \wedge P^{\kappa}_{\overline{\varepsilon}}(y) \wedge P^{\kappa}_{\equiv}(x, y)$

5. $x \not\equiv_{\kappa^{\#}} y$ if $Cond(\kappa) \wedge (\overline{P^{\kappa}_{\overline{\varepsilon}}(x)} \vee \overline{P^{\kappa}_{\overline{\varepsilon}}(y)} \vee \overline{P^{\kappa}_{\equiv}(x,y)})$

where for formulas φ in the language of Peano arithmetic enriched with predicate symbols $\overline{\varepsilon}$, $\not\overline{\varepsilon}$, $\equiv$ and $\not\equiv$ and without any occurrence of $\rightarrow$, the formula $\overline{\varphi}$ represents *the negation of* ϕ and is defined according to the following clauses:

1. for primitive formulas ψ of Peano arithmetic $\overline{\psi} := \neg\psi$

2. $\overline{t\,\overline{\varepsilon}\,u}$ is $t \not\overline{\varepsilon}\ u$, $\overline{t \not\overline{\varepsilon}\ u}$ is $t\,\overline{\varepsilon}\,u$, $\overline{t \equiv_u s}$ is $t \not\equiv_u s$ and $\overline{t \not\equiv_u s}$ is $t \equiv_u s$

3. $\overline{\varphi \wedge \varphi'}$ is $\overline{\varphi} \vee \overline{\varphi'}$ and $\overline{\varphi \vee \varphi'}$ is $\overline{\varphi} \wedge \overline{\varphi'}$

4. $\overline{\exists x\,\varphi}$ is $\forall x\,\overline{\varphi}$ and $\overline{\forall x\,\varphi}$ is $\exists x\,\overline{\varphi}$

Finally, the extra formulas $\kappa^{\#}$, $Cond(\kappa)$, $P^{\kappa}_{\overline{\varepsilon}}$ and $P^{\kappa}_{\equiv}$ are defined in the following tables. $\kappa^{\#}$ makes explicit the encoding of sets built using the constructor κ. $Cond(\kappa)$ is intended to give the constraints which must be respected to define a set through the constructor κ and finally $P^{\kappa}_{\overline{\varepsilon}}$ and $P^{\kappa}_{\equiv}$ give the clauses for membership and equality in a set obtained through the constructor κ, respectively.

κ	$\kappa^{\#}$	$Cond(\kappa)$	$P^{\kappa}_{\overline{\varepsilon}}(x)$
N_0 $\perp$	$p(1,0)$ $p(6,0)$	$\top$	$\perp$
N_1	$p(1,1)$	$\top$	$x = 0$
N	$p(1,2)$	$\top$	$x = x$
List	$p(5,a)$	$Set(a)$	$\forall i\,(i \geq lh(x) \vee (x)_i\,\overline{\varepsilon}\,a)$
$\wedge$	$p(7,p(a,b))$	$Set(a) \wedge Set(b) \wedge$ $p_1(a) > 5 \wedge p_1(b) > 5$	$p_1(x)\,\overline{\varepsilon}\,a \wedge p_2(x)\,\overline{\varepsilon}\,b$
Σ $\exists$	$p(2,p(a,b))$ $p(10,p(a,b))$	$Fam(b,a)$ $Fam_p(b,a)$	$p_1(x)\,\overline{\varepsilon}\,a \wedge p_2(x)\,\overline{\varepsilon}\,\{b\}(p_1(x))$

κ	$\kappa^{\#}$	$Cond(\kappa)$	$P^{\kappa}_{\overline{\varepsilon}}(x)$
$\to$	p(9,p(a,b))	$Set(a)\ \wedge\ Set(b) \wedge$ $p_1(a) > 5 \wedge p_1(b) > 5$	$\forall t\,(t \not\overline{\varepsilon}\ a\ \vee\ \{x\}(t)\,\overline{\varepsilon}\,b)$
Π $\forall$	$p(3, p(a, b))$ $p(11, p(a, b))$	$Fam(b, a)$ $Fam_p(b, a)$	$\forall t\,(t \not\overline{\varepsilon}\ a\ \vee\ \{x\}(t)\,\overline{\varepsilon}\,\{b\}(t)) \wedge$ $\forall t \forall s(t \not\equiv_a s \vee \{x\}(t) \equiv_{\{b\}(t)} \{x\}(s))$
$+$ $\vee$	$p(4, p(a, b))$ $p(8, p(a, b))$	$Set(a)\ \wedge\ Set(b)$ $Set(a)\ \wedge\ Set(b) \wedge$ $p_1(a) > 5 \wedge p_1(b) > 5$	$(p_1(x) = 0\ \wedge\ p_2(x)\,\overline{\varepsilon}\,a) \vee$ $(p_1(x) = 1\ \wedge\ p_2(x)\,\overline{\varepsilon}\,b)$
Eq	$p(12, p(a, p(b, c)))$	$Set(a)\ \wedge\ b\,\overline{\varepsilon}\,a\ \wedge\ c\,\overline{\varepsilon}\,a$	$b \equiv_a c$

κ	$P^{\kappa}_{\equiv}(x)$
N_0	$\perp$
N_1, N	$x = y$
List	$lh(x) = lh(y)\ \wedge\ \forall i\,(i \geq lh(x)\ \vee\ (x)_i\ \equiv_a\ (y)_i)$
Σ	$p_1(x) \equiv_a p_1(y)\ \wedge\ p_2(x) \equiv_{\{b\}(p_1(x))} p_2(y)$
Π	$\forall t\,(t \not\overline{\varepsilon}\ a\ \vee\ \{x\}(t) \equiv_{\{b\}(t)} \{y\}(t))$
$+$	$p_1(x) = p_1(y) \wedge$ $((p_1(x) = 0\ \wedge\ p_2(x) \equiv_a p_2(y))\ \vee\ (p_1(x) = 1\ \wedge\ p_2(x) \equiv_b p_2(y)))$
$\perp, \wedge,$ $\vee, \to,$ $\exists, \forall$, Eq	$\top$

Note that, as already anticipated, families x of small propositions are characterized as those families of sets satisfying the condition $p_1(x) > 5$.

To be more precise, first we define an admissible formula $\varphi(x, X)$ as follows

$$
\begin{aligned}
&\varphi(x,X) \equiv^{def} \bigvee\nolimits_{\kappa} \exists a\, \exists b\, \exists c\, (\\
&\qquad (x = p(20,\kappa) \wedge \mathbf{Cond}(\kappa))\vee\\
&\qquad \exists y\, (x = p(21, p(y,\kappa^{\#})) \wedge \mathbf{Cond}(\kappa) \wedge \mathbf{P}^{\kappa}_{\overline{\varepsilon}}(y))\vee\\
&\qquad \exists y\, (x = p(22, p(y,\kappa^{\#})) \wedge \mathbf{Cond}(\kappa) \wedge \overline{\mathbf{P}^{\kappa}_{\overline{\varepsilon}}(y)})\vee\\
&\qquad \exists y\, \exists z\, (x = p(23, p(\kappa^{\#}, p(y,z))) \wedge \mathbf{Cond}(\kappa) \wedge \mathbf{P}^{\kappa}_{\overline{\varepsilon}}(y) \wedge \mathbf{P}^{\kappa}_{\overline{\varepsilon}}(z) \wedge \mathbf{P}^{\kappa}_{\overline{\equiv}}(y,z))\vee\\
&\qquad \exists y\, \exists z\, (x = p(24, p(\kappa^{\#}, p(y,z))) \wedge \mathbf{Cond}(\kappa) \wedge (\overline{\mathbf{P}^{\kappa}_{\overline{\varepsilon}}(y)} \vee \overline{\mathbf{P}^{\kappa}_{\overline{\varepsilon}}(z)} \vee \overline{\mathbf{P}^{\kappa}_{\overline{\equiv}}(y,z)})))
\end{aligned}
$$

where the disjunction $\bigvee_{\kappa}$ is the finite disjunction indexed by the constructors κ in the previous tables and where the boldface versions of $Cond(\kappa)$, $P^{\kappa}_{\overline{\varepsilon}}(x)$ and $P^{\kappa}_{\overline{\equiv}}(y,z)$ are obtained by substituting $Set(u)$, $t\,\overline{\varepsilon}\,u$, $t \not\overline{\varepsilon}\, u$, $s \equiv_u t$ and $s \not\equiv_u t$ with $p(20,u)\,\varepsilon\, X$, $p(21, p(t,u))\,\varepsilon\, X$, $p(22, p(t,u))\,\varepsilon\, X$, $p(23, p(u, p(s,t)))\,\varepsilon\, X$ and $p(24, p(u, p(s,t)))\,\varepsilon\, X$ respectively in the original formulas.

For the sake of example let us show what is the subformula of $\varphi(x, X)$ corresponding to κ equal to N:

$$
\begin{aligned}
&\exists a\, \exists b\, \exists c\, ((x = p(20, p(1,2)) \wedge \top)\vee\\
&\exists y\, (x = p(21, p(y, p(1,2))) \wedge \top \wedge y = y)\vee\\
&\exists y\, (x = p(22, p(y, p(1,2))) \wedge \top \wedge \neg\, y = y)\vee\\
&\exists y\, \exists z\, (x = p(23, p(p(1,2), p(y,z))) \wedge \top \wedge y = y \wedge z = z \wedge y = z)\vee\\
&\exists y\, \exists z\, (x = p(24, p(p(1,2), p(y,z))) \wedge \top \wedge (\neg\, y = y \vee \neg\, z = z \vee \neg\, y = z)))
\end{aligned}
$$

Then we consider the fixpoint formula $P_{\varphi}(x)$ corresponding to $\varphi(x, X)$ and we define

1. $Set(x) \equiv^{def} P_{\varphi}(p(20,x))$
2. $x\,\overline{\varepsilon}\,y \equiv^{def} P_{\varphi}(p(21, p(x,y)))$
3. $x \not\overline{\varepsilon}\, y \equiv^{def} P_{\varphi}(p(22, p(x,y)))$
4. $x \equiv_z y \equiv^{def} P_{\varphi}(p(23, p(z, p(x,y))))$
5. $x \not\equiv_z y \equiv^{def} P_{\varphi}(p(24, p(z, p(x,y))))$

In this framework we can define the following formulas:

- $Coh(x) \equiv^{def} \forall t\,(t\,\overline{\varepsilon}\,x \leftrightarrow \neg(t \not\overline{\varepsilon}\, x)) \wedge \forall t \forall s\,(t \equiv_x s \leftrightarrow \neg(t \not\equiv_x s))$ stating that the formulas $t \not\overline{\varepsilon}\, x$ and $t \not\equiv_x s$ defined by fixpoint behave really like negations of $t\,\overline{\varepsilon}\,x$ and $t \equiv_x s$ respectively;
- $Wd(x) \equiv^{def} \forall t \forall s\,(t \equiv_x s \rightarrow t\,\overline{\varepsilon}\,x \wedge s\,\overline{\varepsilon}\,x)$ stating that the relation $t \equiv_x s$ is well defined on x;
- $Ref(x) \equiv^{def} \forall t\,(t\,\overline{\varepsilon}\,x \leftrightarrow t \equiv_x t)$ stating that the relation $t \equiv_x s$ is reflexive on x;
- $Sym(x) \equiv^{def} \forall t \forall s\,(t \equiv_x s \rightarrow s \equiv_x t)$ stating that the relation $t \equiv_x s$ is symmetric;
- $Tra(x) \equiv^{def} \forall t \forall s \forall u\,(t \equiv_x s \wedge s \equiv_x u \rightarrow t \equiv_x u)$ stating that the relation $t \equiv_x s$ is transitive;
- $EqR(x) \equiv^{def} Ref(x) \wedge Sym(x) \wedge Tra(x)$ stating that the relation $t \equiv_x s$ is an equivalence relation on x;
- $PrIrr(x) \equiv^{def} \forall t \forall s\,(t\,\overline{\varepsilon}\,x \wedge s\,\overline{\varepsilon}\,x \leftrightarrow t \equiv_x s)$ stating that the relation $t \equiv_x s$ is trivial on x (i. e. x is proof-irrelevant);
- $x =_{ext} y \equiv^{def} \forall t \forall s\,(t \equiv_x s \leftrightarrow t \equiv_y s)$ stating that two sets are defined *extensionally equal* if they share the same equivalent (equal) elements.

Notice that the following hold:

- $\widehat{ID_1} \vdash Ref(x) \wedge Ref(y) \wedge x =_{ext} y \rightarrow \forall t\,(t\,\overline{\varepsilon}\,x \leftrightarrow t\,\overline{\varepsilon}\,y)$ namely, two reflexive sets are *extensionally equal* if and only if they *share the same elements*;
- $\widehat{ID_1} \vdash PrIrr(x) \rightarrow EqR(x)$
- $\widehat{ID_1} \vdash EqR(x) \rightarrow Wd(x)$

4.5.1 The classifier of realized sets and that of small realized propositions

Definition 4.57. *We define the collection* US *as the universe of codes for sets with extensional equality:*

- $|\mathsf{US}| := \{x \mid Set(x) \wedge Coh(x) \wedge EqR(x)\}$

- $x \sim_{\mathsf{US}} y$ *is* $x \,\varepsilon\, \mathsf{US} \wedge y \,\varepsilon\, \mathsf{US} \wedge x =_{ext} y$

Definition 4.58. *We define the collection* USP *as the universe of codes for small propositions:*

- $|\mathsf{USP}| := \{x \,|\, Set(x) \wedge p_1(x) > 5 \wedge Coh(x) \wedge PrIrr(x)\}$
- $x \sim_{\mathsf{USP}} y$ *is* $x \,\varepsilon\, \mathsf{USP} \wedge y \,\varepsilon\, \mathsf{USP} \wedge x =_{ext} y$

Definition 4.59. *For every object* Γ *in* **Cont** *we define the following families of collections in* $\mathbf{Col}(\Gamma)$*:*

$$\mathsf{US}^{\Gamma} := \mathbf{Col}_{!_{\Gamma,[\,]}}(\mathsf{US}) \qquad\qquad \mathsf{USP}^{\Gamma} := \mathbf{Col}_{!_{\Gamma,[\,]}}(\mathsf{USP})$$

Definition 4.60. *For any object* Γ *of* **Cont** *we define* τ^{Γ} *as the collection depending on* $\overline{x}_{\ell(\Gamma)+1}$ *determined by the following conditions:*

- $x \;\varepsilon\; \tau^{\Gamma}$ *is* $x \;\overline{\varepsilon}\; x_{\ell(\Gamma)+1} \wedge \overline{x}_{|_{\ell(\Gamma)}} \,\varepsilon\, \Gamma \wedge x_{\ell(\Gamma)+1} \,\varepsilon\, \mathsf{US}$
- $x \sim_{\tau^{\Gamma}} y$ *is* $x \equiv_{x_{\ell(\Gamma)+1}} y \wedge \overline{x}_{|_{\ell(\Gamma)}} \,\varepsilon\, \Gamma \wedge x_{\ell(\Gamma)+1} \,\varepsilon\, \mathsf{US}$

Lemma 4.61. *Suppose* Γ *is an object of* **Cont**. *Then* τ^{Γ} *is an object of* $\mathbf{Col}([\Gamma, \mathsf{US}^{\Gamma}])$ *and the following are well defined arrows in* $\mathbf{Col}(\Gamma)$*:*

- $\widehat{\mathsf{N}_0}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(1,0)}$ *and* $\widehat{\mathsf{N}_1}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(1,1)}$ *and* $\widehat{\mathsf{N}}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(1,2)}$ *from* $\mathbf{1}^{\Gamma}$ *to* US^{Γ}
- $\widehat{\Sigma}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(2,x)}$ *and* $\widehat{\Pi}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(3,x)}$ *from* $\Sigma^{\Gamma}(\mathsf{US}^{\Gamma}, \tau^{\Gamma} \Rightarrow^{[\Gamma,\mathsf{US}^{\Gamma}]} \mathsf{US}^{[\Gamma,\mathsf{US}^{\Gamma}]})$ *to* US^{Γ}
- $\widehat{+}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(4,x)}$ *from* $\mathsf{US}^{\Gamma} \times^{\Gamma} \mathsf{US}^{\Gamma}$ *to* US^{Γ}
- $\widehat{\mathsf{List}}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(5,x)}$ *from* US^{Γ} *to* US^{Γ}
- $\widehat{\perp}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(6,0)} : \mathbf{1}^{\Gamma} \to \mathsf{USP}^{\Gamma}$
- $\widehat{\wedge}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(7,x)}$ *and* $\widehat{\vee}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(8,x)}$ *and* $\widehat{\Rightarrow}^{\Gamma} := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(9,x)}$ *from* $\mathsf{USP}^{\Gamma} \times^{\Gamma} \mathsf{USP}^{\Gamma}$ *to* USP^{Γ}

- $\widehat{\exists}^\Gamma := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(10,x)}$ *and* $\quad \widehat{\forall}^\Gamma := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(11,x)}$
  from $\Sigma^\Gamma(\mathsf{US}^\Gamma, \tau^\Gamma \Rightarrow^{[\Gamma,\mathsf{US}^\Gamma]} \mathsf{USP}^{[\Gamma,\mathsf{US}^\Gamma]})$ *to* USP^Γ
- $\widehat{\mathsf{Eq}}^\Gamma := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.\{\mathbf{p}\}(12,x)}$
  from $\Sigma^\Gamma(\mathsf{US}^\Gamma, \tau^\Gamma \times^{[\Gamma,\mathsf{US}^\Gamma]} \tau^\Gamma)$ *to* USP^Γ
- $\sigma^\Gamma := \gamma_{\Lambda \overline{x}_{|_{\ell(\Gamma)}}.\Lambda x.x}$ *from* USP^Γ *to* US^Γ

4.6 Dependent realized sets and small realized propositions and their indexed categories

Here we finally give the definitions of realized sets and of small realized propositions by using their classifiers.

Note that, for any context Γ, any generalized element of US over Γ (i. e. an arrow from Γ to US) in **Cont**, or equivalently any global element of US^Γ in $\mathbf{Col}(\Gamma)$ gives rise to a realized collection in $\mathbf{Col}(\Gamma)$:

Lemma 4.62. *Let* Γ *be an object of* **Cont** *with* $\ell(\Gamma) = n$.

Suppose $f = \gamma_{\mathbf{n}_f} : \mathbf{1}^\Gamma \to \mathsf{US}^\Gamma$ *(recall the notation in lemma 4.16) in* $\mathbf{Col}(\Gamma)$*. Then the collection* $\tau_s^\Gamma(f)$ *of* $\widehat{ID_1}$ *depending on* $\overline{x}_{|_n}$ *defined by*

1. $x \,\varepsilon\, \tau_s^\Gamma(f) \equiv^{def} \overline{x}_{|_n} \,\varepsilon\, \Gamma \wedge x \,\overline{\varepsilon}\, \{\mathbf{n}_f\}(\overline{x}_{|_n}, 0)$
2. $x \sim_{\tau_s^\Gamma(f)} y \equiv^{def} \overline{x}_{|_n} \,\varepsilon\, \Gamma \wedge x \equiv_{\{\mathbf{n}_f\}(\overline{x}_{|_n},0)} y$

is a well defined object of $\mathbf{Col}(\Gamma)$*. Moreover, for arrows* $f, g : \mathbf{1}^\Gamma \to \mathsf{US}^\Gamma$ *in* $\mathbf{Col}(\Gamma)$*, if* $\tau_s^\Gamma(f)$ *is equal to* $\tau_s^\Gamma(g)$*, then* f *and* g *are equal arrows in* $\mathbf{Col}(\Gamma)$*.*

Note that any global element of USP^Γ in $\mathbf{Col}(\Gamma)$, or equivalently any generalized element of USP over Γ in **Cont**, gives rise to a realized proposition in $\mathbf{Prop}(\Gamma)$:

Lemma 4.63. *Let* Γ *be an object of* **Cont** *with* $\ell(\Gamma) = n$.

Suppose $f = \gamma_{\mathbf{n}_f} : \mathbf{1}^\Gamma \to \mathsf{USP}^\Gamma$*, then the collection* $\tau_{sp}^\Gamma(f)$ *of* $\widehat{ID_1}$ *depending on* $\overline{x}_{|_n}$ *defined by*

1. $x \,\varepsilon\, \tau_{sp}^\Gamma(f) \equiv^{def} \overline{x}_{|_n} \,\varepsilon\, \Gamma \wedge x \,\overline{\varepsilon}\, \{\mathbf{n}_f\}(\overline{x}_{|_n}, 0)$
2. $x \sim_{\tau_{sp}^\Gamma(f)} y \equiv^{def} \overline{x}_{|_n} \,\varepsilon\, \Gamma \wedge x \equiv_{\{\mathbf{n}_f\}(\overline{x}_{|_n},0)} y$

is a well defined object of $\mathbf{Prop}(\Gamma)$*. Moreover, for arrows* $f, g : \mathbf{1}^\Gamma \to \mathsf{USP}^\Gamma$ *in* $\mathbf{Col}(\Gamma)$*, if* $\tau_{sp}^\Gamma(f)$ *is equal to* $\tau_{sp}^\Gamma(g)$*, then* f *and* g *are equal arrows in* $\mathbf{Col}(\Gamma)$*.*

Finally, we define a realized set depending on a context Γ as the collection of elements of a global element of US^Γ in $\mathbf{Col}(\Gamma)$ or, equivalently, of a generalized element of US over Γ in $\mathbf{Cont}$:

Definition 4.64. *If Γ is an object of $\mathbf{Cont}$, a* realized set depending on Γ *(or a* family of realized sets on Γ*) is a realized collection of the form $\tau_s^\Gamma(f)$ for an arrow $f : \mathbf{1}^\Gamma \to \mathsf{US}^\Gamma$ in $\mathbf{Col}(\Gamma)$.*

Analogously, we define a small realized proposition depending on a context Γ as the collection of elements of a global element of USP^Γ in $\mathbf{Col}(\Gamma)$ or, equivalently, of a generalized element of USP over Γ in $\mathbf{Cont}$:

Definition 4.65. *If Γ is an object of $\mathbf{Cont}$, a* small realized proposition depending on Γ *(or a* family of realized small propositions on Γ*) is a realized collection of the form $\tau_{sp}^\Gamma(f)$ for an arrow $f : \mathbf{1}^\Gamma \to \mathsf{USP}^\Gamma$ in $\mathbf{Col}(\Gamma)$.*

Now we are ready to define the indexed category of realized sets and that of small realized propositions. We start by defining their fibres as follows:

Definition 4.66. *If Γ is an object of $\mathbf{Cont}$, we define $\mathbf{Set}(\Gamma)$ as the full subcategory of $\mathbf{Col}(\Gamma)$ whose objects are realized sets depending on Γ.*

Moreover, if A is an object of $\mathbf{Set}(\Gamma)$, we write $\mathbf{en}_s^\Gamma(A)$ for the arrow satisfying $A = \tau_s^\Gamma(\mathbf{en}_s^\Gamma(A))$.

Definition 4.67. *If Γ is an object of $\mathbf{Cont}$, we define $\mathbf{Prop}_s(\Gamma)$ as the full subcategory of $\mathbf{Col}(\Gamma)$ whose objects are small realized propositions depending on Γ.*

The substitution functors for $\mathbf{Set}$ and $\mathbf{Prop}_s$ will be both inherited from those of $\mathbf{Col}$:

Lemma 4.68. *If $f : \Gamma' \to \Gamma$ in $\mathbf{Cont}$ and A is an object of $\mathbf{Set}(\Gamma)$ (resp. of $\mathbf{Prop}_s(\Gamma)$), then $\mathbf{Col}_f(A)$ is an object of $\mathbf{Set}(\Gamma')$ (resp. of $\mathbf{Prop}_s(\Gamma')$).*

Definition 4.69. *The pair of assignments*

$$\Gamma \mapsto \mathbf{Set}(\Gamma) \qquad f \mapsto \mathbf{Set}_f := \mathbf{Col}_f|_{\mathbf{Set}(\mathsf{cod}(f))}$$

where $\mathsf{cod}(f)$ is the codomain of f, defines an indexed category,

$$\mathbf{Set} : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

Definition 4.70. *The pair of assignments*

$$\Gamma \mapsto \mathbf{Prop}_s(\Gamma) \qquad f \mapsto \mathbf{Prop}_{s,f} := \mathbf{Col}_f|_{\mathbf{Prop}_s(\mathsf{cod}(f))}$$

where $\mathsf{cod}(f)$ is the codomain of f, defines an indexed category

$$\mathbf{Prop}_s : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

The indexed category of small realized propositions is a sub-indexed category of that of realized sets:

Lemma 4.71. *If* Γ *is an object of* **Cont**, *every object in* $\mathbf{Prop}_s(\Gamma)$ *is also in* $\mathbf{Set}(\Gamma)$.

The following lemma is instrumental to show that the fibres of **Set** and of $\mathbf{Prop}_s$ are closed under finite limits, finite coproducts, function spaces and under left and right adjoints to substitution functors along morphisms of the kind $\mathsf{pr}_{[\Gamma,A]}$ for any A in $\mathbf{Set}(\Gamma)$ (they are not closed under left and right adjoints to substitution functors along any morphism of **Cont** for predicativity reasons!). Moreover, from this lemma it also follows that each fibre of **Set** is closed under list objects and contains the natural numbers object of the corresponding fibre of **Col**.

Lemma 4.72. *Let* Γ *be an object of* **Cont**. *Then*

1. $\perp^{\Gamma}$ *is in* $\mathbf{Prop}_s(\Gamma)$*;*
2. $\mathbf{0}^{\Gamma}$, $\mathbf{1}^{\Gamma}$ *and* $\mathbf{N}^{\Gamma}$ *are in* $\mathbf{Set}(\Gamma)$*;*
3. *if* A *and* B *are in* $\mathbf{Set}(\Gamma)$, *then* $A \times^{\Gamma} B$, $A +^{\Gamma} B$ *and* $A \Rightarrow^{\Gamma} B$ *are in* $\mathbf{Set}(\Gamma)$*;*
4. *if* A *is in* $\mathbf{Set}(\Gamma)$, *then* $\mathbf{List}^{\Gamma}(A)$ *is in* $\mathbf{Set}(\Gamma)$*;*
5. *if* A *is in* $\mathbf{Set}(\Gamma)$ *and* B *is in* $\mathbf{Set}([\Gamma, A])$, *then* $\Pi^{\Gamma}(A, B)$ *and* $\Sigma^{\Gamma}(A, B)$ *are in* $\mathbf{Set}(\Gamma)$*;*
6. *if* P *and* Q *are in* $\mathbf{Prop}_s(\Gamma)$, *then* $P \sqcup^{\Gamma} Q$, $P \sqcap^{\Gamma} Q$ *and* $P \to^{\Gamma} Q$ *are in* $\mathbf{Prop}_s(\Gamma)$*;*
7. *if* A *is in* $\mathbf{Set}(\Gamma)$ *and* P *is in* $\mathbf{Prop}_s([\Gamma, A])$, *then* $\forall^{\Gamma}(A, P)$ *and* $\exists^{\Gamma}(A, P)$ *are in* $\mathbf{Prop}_s(\Gamma)$*;*
8. *if* A *is in* $\mathbf{Set}(\Gamma)$, *for arrows* $f, g : \mathbf{1}^{\Gamma} \to A$ *in* $\mathbf{Col}(\Gamma)$, *then* $\mathbf{Eq}^{\Gamma}(A, f, g)$ *is in* $\mathbf{Prop}_s(\Gamma)$.

The following lemma will be useful to validate the equality rules of the collection of small propositions in $\mathbf{mTT}^a$.

Lemma 4.73. *Let* Γ *be an object of* **Cont**, *let* $f : \mathbf{1} \to \mathsf{US}^{\Gamma}$, $p, p' : \mathbf{1} \to \mathsf{USP}^{\Gamma}$ *and* $g, g' : \mathbf{1} \to \tau_s^{\Gamma}(f)$ *be arrows in* $\mathbf{Col}(\Gamma)$ *and let* $h : \mathbf{1} \to \mathsf{USP}^{[\Gamma, \tau_s^{\Gamma}(f)]}$ *be an arrow in* $\mathbf{Col}([\Gamma, \tau_s^{\Gamma}(f)])$. *Then in* $\mathbf{Prop}_s(\Gamma)$ *(recall the notation in lemma 4.40):*

1. $\tau_{sp}^{\Gamma}(\widehat{\perp}^{\Gamma})$ *coincides with* $\perp^{\Gamma}$*;*

2. $\tau_{sp}^{\Gamma}(\widehat{\wedge}^{\Gamma} \circ \langle p, p' \rangle)$ *coincides with* $\tau_{sp}^{\Gamma}(p) \sqcap^{\Gamma} \tau_{sp}^{\Gamma}(p')$*;*

3. $\tau_{sp}^{\Gamma}(\widehat{\vee}^{\Gamma} \circ \langle p, p' \rangle)$ *coincides with* $\tau_{sp}^{\Gamma}(p) \sqcup^{\Gamma} \tau_{sp}^{\Gamma}(p')$*;*

4. $\tau_{sp}^{\Gamma}(\widehat{\rightarrow}^{\Gamma} \circ \langle p, p' \rangle)$ *coincides with* $\tau_{sp}^{\Gamma}(p) \rightarrow^{\Gamma} \tau_{sp}^{\Gamma}(p')$*;*

5. $\tau_{sp}^{\Gamma}(\widehat{\exists}^{\Gamma} \circ \langle f, \mathsf{Cur}(h_{\natural}^{\Gamma} \circ \pi_2^{\mathbf{1}, \tau_s^{\Gamma}(f)}) \rangle_{\Sigma})$ *coincides with* $\exists^{\Gamma}(\tau_s^{\Gamma}(f), \tau_{sp}^{[\Gamma, \tau_s^{\Gamma}(f)]}(h))$*;*

6. $\tau_{sp}^{\Gamma}(\widehat{\forall}^{\Gamma} \circ \langle f, \mathsf{Cur}(h_{\natural}^{\Gamma} \circ \pi_2^{\mathbf{1}, \tau_s^{\Gamma}(f)}) \rangle_{\Sigma})$ *coincides with* $\forall^{\Gamma}(\tau_s^{\Gamma}(f), \tau_{sp}^{[\Gamma, \tau_s^{\Gamma}(f)]}(h))$*;*

7. $\tau_{sp}^{\Gamma}(\widehat{\mathsf{Eq}}^{\Gamma} \circ \langle f, \langle g, g' \rangle \rangle_{\Sigma})$ *coincides with* $\mathsf{Eq}(\tau_s^{\Gamma}(f), g, g')$*;*

4.7 Structure of the effective pretripos

What shown so far, together with well known results in categorical logic (see [32], [37], [26]), allows to prove the following:

Theorem 4.74. *The functor*

$$\mathbf{Col} : \mathbf{Cont}^{op} \rightarrow \mathbf{Cat}$$

is an indexed category whose fibres $\mathbf{Col}(\Gamma)$ *for any* Γ *in* $\mathbf{Cont}$ *are finitely complete cartesian closed categories with finite coproducts and list objects. Moreover, for any morphism* f *in* $\mathbf{Cont}$ *the substitution functor* $\mathbf{Col}_f$ *preserves the mentioned fibre structure and it has both left and right adjoints satisfying Beck-Chevalley conditions. Finally, the fibre on the terminal object* $\mathbf{Col}([\,])$ *is equivalent to* $\mathbf{Cont}$ *itself making it a locally cartesian closed category.*

The functor

$$\mathbf{Prop} : \mathbf{Cont}^{op} \rightarrow \mathbf{Cat}$$

is an indexed full subcategory of $\mathbf{Col}$ *which is also a hyperdoctrine according to the notion defined in [37], and its posetal reflection it is a first order hyperdoctrine in the sense of [32].*

The functor

$$\mathbf{Set} : \mathbf{Cont}^{op} \rightarrow \mathbf{Cat}$$

is an indexed full subcategory of $\mathbf{Col}$ *whose fibres are also finitely complete cartesian closed categories with finite coproducts and list objects. Moreover, for any morphism* f *in* $\mathbf{Cont}$*, the substitution functor* $\mathbf{Set}_f$ *preserves the mentioned fibre structure and, for any* $f = \mathsf{pr}_{[\Gamma, A]}$ *with* A *in* $\mathbf{Set}(\Gamma)$*, it has both left and right adjoints satisfying Beck-Chevalley conditions.*

The functor

$$\mathbf{Prop}_s : \mathbf{Cont}^{op} \to \mathbf{Cat}$$

is an indexed full subcategory both of $\mathbf{Prop}$ *and of* $\mathbf{Set}$ *whose fibres are Heyting pre-algebras. Moreover, for any morphism* f *in* $\mathbf{Cont}$*, the substitution functor* $(\mathbf{Prop}_s)_f$ *preserves the mentioned fibre structure and for any* $f = \mathsf{pr}_{[\Gamma,A]}$ *with* A *in* $\mathbf{Set}(\Gamma)$ *it has both left and right adjoints satisfying Beck-Chevalley conditions.*

Furthermore, for every object Γ *in* $\mathbf{Cont}$ *the object* US *allows to represent the functor* $\mathbf{Set}$ *in the sense that there is a bijection*

$$\mathbf{Cont}(\Gamma, \mathsf{US}) \simeq \mathbf{Set}(\Gamma)$$

natural in Γ *and the object* USP *allows to represent* $\mathbf{Prop}_s$ *in the sense that there is a bijection*

$$\mathbf{Cont}(\Gamma, \mathsf{USP}) \simeq \mathbf{Prop}_s(\Gamma)$$

natural in Γ*.*

Finally all the embeddings in the below diagram preserve the relevant mentioned structures of each indexed category:

$$\begin{tikzcd}
\mathbf{Set} \arrow[r, hook] & \mathbf{Col} \\
\mathbf{Prop}_s \arrow[u, hook] \arrow[r, hook] & \mathbf{Prop} \arrow[u, hook]
\end{tikzcd}$$

Definition 4.75. *The 5-tuple* $(\mathbf{Cont}, \mathbf{Col}, \mathbf{Set}, \mathbf{Prop}, \mathbf{Prop}_s)$ *is called the* effective pretripos for $\mathbf{mTT}$*.*

We will see later that the principle of formal Church thesis will be validated in the effective pretripos for $\mathbf{mTT}$.

5 The interpretation of the Minimalist Foundation

Here we give a partial interpretation $\mathcal{I}$ of precontexts and of types and terms in precontext of the fully annotated syntax of $\mathbf{mTT}^a$ in our effective pretripos for $\mathbf{mTT}$ following Streicher's technique in [38]. We call the resulting model $\mathcal{R}$.

Definition 5.1. *The validity of judgements in the model* ($\mathcal{R} \vDash J$) *is defined as follows:*

$\mathcal{R} \vDash$	*if*
$\Gamma\, context$	$\mathcal{I}(\Gamma)$ *is well defined and is an object of* **Cont**
$B\, col\, [\Gamma]$	$\mathcal{R} \vDash \Gamma\, context$ *and* $\mathcal{I}(B\,[\Gamma])$ *is a well defined object of* $\mathbf{Col}(\mathcal{I}(\Gamma))$
$A\, set\, [\Gamma]$	$\mathcal{R} \vDash A\, col\, [\Gamma]$ *and* $\mathcal{I}(A\,[\Gamma])$ *is an object of* $\mathbf{Set}(\mathcal{I}(\Gamma))$
$\phi\, prop\, [\Gamma]$	$\mathcal{R} \vDash \phi\, col\, [\Gamma]$ *and* $\mathcal{I}(\phi\,[\Gamma])$ *is an object of* $\mathbf{Prop}(\mathcal{I}(\Gamma))$
$\phi\, prop_s\, [\Gamma]$	$\mathcal{R} \vDash \phi\, col\, [\Gamma]$ *and* $\mathcal{I}(\phi\,[\Gamma])$ *is an object of* $\mathbf{Prop}_s(\mathcal{I}(\Gamma))$
$A = B\ type\, [\Gamma]$ $type \in$ $\{col, set, prop, prop_s\}$	$\mathcal{R} \vDash A\, type$ *and* $\mathcal{R} \vDash B\, type$ *and* $\mathcal{I}(A[\Gamma])$ *and* $\mathcal{I}(B[\Gamma])$ *are equal objects of* $\mathbf{Col}(\mathcal{I}(\Gamma))$
$a \in A\, [\Gamma]$	$\mathcal{R} \vDash A\, col\, [\Gamma]$ *and* $\mathcal{I}(a[\Gamma])$ *is well defined* *and* $\mathcal{I}(a[\Gamma]) : \mathbf{1}^{\mathcal{I}(\Gamma)} \to \mathcal{I}(A[\Gamma])$ *is in* $\mathbf{Col}(\mathcal{I}(\Gamma))$
$a = b \in A\, [\Gamma]$	$\mathcal{R} \vDash a \in A$ *and* $\mathcal{R} \vDash b \in A$ *and* $\mathcal{I}(a[\Gamma])$ *and* $\mathcal{I}(b[\Gamma])$ *are equal arrows of* $\mathbf{Col}(\mathcal{I}(\Gamma))$

Definition 5.2. *If* $\mathbf{mTT}^a \vdash \phi\, prop\, [\Gamma]$ *we will say that* $\mathcal{R}$ *validates* ϕ *in context* Γ*, also written* $\mathcal{R} \vDash \phi\, [\Gamma]$*, if we have that* $\mathcal{R} \vDash \phi\, prop\, [\Gamma]$ *and* $\top^{\mathcal{I}(\Gamma)} \sqsubseteq^{\mathcal{I}(\Gamma)} \mathcal{I}(\phi[\Gamma])$ *in* $\mathbf{Prop}(\mathcal{I}(\Gamma))$.

In the next subsections we will omit superscripts and subscripts in the categorical notation when they will be clear from the context.

5.1 Precontexts

We interpret precontexts as objects of **Cont** as follows:

$\mathcal{I}([\,]) := [\,] \in Ob(\mathbf{Cont})$;

$\mathcal{I}([\Gamma, x \in A]) := [\mathcal{I}(\Gamma), \mathcal{I}(A[\Gamma])]$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\Gamma)$.

5.2 Variables

If $\Gamma := [x_1 \in A_1, ..., x_n \in A_n]$, then variables in context are defined as arrows in $\mathbf{Col}(\Gamma)$ as follows

$$\mathbf{1} \xrightarrow{\mathcal{I}(x_1 \in A_1[\Gamma]) := \gamma_{\pi_1^{n+1}}} \mathbf{Col}_{\mathsf{pr}^{(n)}_{\mathcal{I}(\Gamma)}}(\mathcal{I}(A_1[\,]))$$

$$\mathbf{1} \xrightarrow{\mathcal{I}(x_{i+1} \in A_{i+1}[\Gamma]) := \gamma_{\pi_{i+1}^{n+1}}} \mathbf{Col}_{\mathsf{pr}^{(n-i)}_{\mathcal{I}(\Gamma)}}(\mathcal{I}(A_{i+1}[x_1 \in A_1, ..., x_i \in A_i]))$$

if $1 \leq i \leq n-1$.

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**.

5.3 Basic sets

We interpret the emptyset, the singleton and the natural numbers type as follows:

$$\mathcal{I}(\mathsf{N}_0[\Gamma]) := \mathbf{0}^{\mathcal{I}(\Gamma)} \qquad \mathcal{I}(\mathsf{N}_1[\Gamma]) := \mathbf{1}^{\mathcal{I}(\Gamma)} \qquad \mathcal{I}(\mathsf{N}[\Gamma]) := \mathbf{N}^{\mathcal{I}(\Gamma)}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**.

The interpretation of the emptyset eliminator $\mathsf{emp}_0^A(a)[\Gamma]$ is defined as the composed arrow in the following commuting diagram in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "\mathcal{I}(a[\Gamma])"] \arrow[dr, "\mathcal{I}(\mathsf{emp}_0^A(a)[\Gamma])"'] & \mathbf{0} \arrow[d, "!_{\mathbf{0},\mathcal{I}(A[\Gamma])}"] \\
& \mathcal{I}(A[\Gamma])
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{0}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the singleton constant $\star[\Gamma]$ is $\mathcal{I}(\star[\Gamma]) := id_{\mathbf{1}} : \mathbf{1} \to \mathbf{1}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$, provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**.

The interpretation of the singleton eliminator $\mathsf{El}_{\mathsf{N}_1}^A(b, a)[\Gamma]$ is defined as the composed arrow in the following commuting diagram in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "\mathcal{I}(b[\Gamma])"] \arrow[dr, "\mathcal{I}(\mathsf{El}_{\mathsf{N}_1}^A(b,a)[\Gamma])"'] & \mathbf{1} \arrow[d, "\mathcal{I}(a[\Gamma])"] \\
& \mathcal{I}(A[\Gamma])
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(b[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{1}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the constant $0[\Gamma]$ is defined as $\mathcal{I}(0[\Gamma]) := \mathsf{z} : \mathbf{1} \to \mathbf{N}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$, provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**.

The interpretation of the successor constructor $\mathsf{succ}(a)[\Gamma]$ is defined as

$$\mathcal{I}(\mathsf{succ}(a)[\Gamma]) := \mathsf{s} \circ \mathcal{I}(a[\Gamma]) : \mathbf{1} \to \mathbf{N}$$

in $\mathbf{Col}(\mathcal{I}(\Gamma))$ according with the notation in remark 4.30, provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{N}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the natural numbers eliminator $\mathsf{El}_{\mathsf{N}}^{A}(a, b, (x)\, c)[\Gamma]$ is defined as

$$\mathcal{I}(\mathsf{El}_{\mathsf{N}}^{A}(a, b, (x)\, c)[\Gamma]) := rec(\,\mathcal{I}(b[\Gamma]), \mathcal{I}(c[\Gamma, x \in A])_{\natural}^{\mathcal{I}(\Gamma)}\,) \circ \mathcal{I}(a[\Gamma]) : \mathbf{1} \to \mathcal{I}(A[\Gamma])$$

in $\mathbf{Col}(\mathcal{I}(\Gamma))$ according with the notation in remark 4.30 and lemma 4.40, provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{N}$ in $\mathbf{Col}(\mathcal{I}([\Gamma]))$, $\mathcal{I}(b[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}([\Gamma]))$ and $\mathcal{I}(c[\Gamma, x \in A])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{Col}_{\mathsf{pr}}(\mathcal{I}(A[\Gamma]))$ in $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$.

5.4 Dependent sums

We interpret the dependent sum as follows:

$$\mathcal{I}((\Sigma x \in A)B[\Gamma]) := \Sigma^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$. The interpretation of the pairing of the dependent sum $\langle a, b\rangle^{A,(x)B}[\Gamma]$ is defined as

$$\mathcal{I}(\langle a, b\rangle^{A,(x)B}[\Gamma]) := \langle \mathcal{I}(a[\Gamma]), \mathcal{I}(b[\Gamma])\rangle_{\Sigma} : \mathbf{1} \to \Sigma^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$$

with reference to lemma 4.34 provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$, $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}([\Gamma]))$ and $\mathcal{I}(b[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{Col}_{\widetilde{\mathcal{I}(a[\Gamma])}}(\,\mathcal{I}(B[\Gamma, x \in A])\,)$ in $\mathbf{Col}(\mathcal{I}([\Gamma]))$.

The interpretations of the projections of the dependent sum $\pi_1^{A,(x)B}(c)[\Gamma]$ and $\pi_2^{A,(x)B}(c)[\Gamma]$ are defined as follows

$$\mathcal{I}(\pi_1^{A,(x)B}(c)[\Gamma]) := \mathsf{p}_1^{\Sigma} \circ \mathcal{I}(c[\Gamma]) : \mathbf{1} \to \mathcal{I}(A[\Gamma])$$

$$\mathcal{I}(\pi_2^{A,(x)B}(c)[\Gamma]) := \mathsf{p}_2^{\Sigma}(\mathcal{I}(c[\Gamma])) : \mathbf{1} \to \mathbf{Col}_{\widetilde{\mathsf{p}_1^{\Sigma} \circ \mathcal{I}(c[\Gamma])}}(\mathcal{I}(B[\Gamma]))$$

with reference to lemma 4.34 provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$ and $\mathcal{I}(c[\Gamma])$ is a well defined arrow in $\mathbf{Col}(\mathcal{I}(\Gamma))$ from $\mathbf{1}$ to $\Sigma^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$.

5.5 Dependent products

We interpret the dependent product as follows:

$$\mathcal{I}((\Pi x \in A)B[\Gamma]) := \Pi^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$.

The interpretation of the lambda-abstraction $(\lambda x)^{A,B}b[\Gamma]$ is defined as

$$\mathcal{I}((\lambda x)^{A,B}b[\Gamma]) := \mathsf{Cur}_{\Pi}(\mathcal{I}(b[\Gamma, x \in A])) : \mathbf{1} \to \Pi^{\mathcal{I}(\Gamma)}(\mathcal{I}(A), \mathcal{I}(B[\Gamma, x \in A]))$$

in $\mathbf{Col}(\mathcal{I}(\Gamma))$ where $\mathsf{Cur}_{\Pi}(\mathcal{I}(b[\Gamma, x \in A]))$ is the arrow (see corollary 4.37) making the following diagram commute in $\mathbf{Col}([\mathcal{I}(\Gamma), \mathcal{I}(A[\Gamma])])$

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "\mathcal{I}(b[\Gamma, x \in A])"] \arrow[d, "\mathbf{Col}_{\mathsf{pr}}(\mathsf{Cur}_{\Pi}(\mathcal{I}(b[\Gamma, x \in A])))"'] & \mathcal{I}(B[\Gamma, x \in A]) \\
\mathbf{Col}_{\mathsf{pr}}(\Pi^{\mathcal{I}(\Gamma)}(\mathcal{I}(A), \mathcal{I}(B[\Gamma, x \in A]))) \arrow[ur, "\mathsf{ev}_{\Pi}^{\Gamma}"'] &
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$ and $\mathcal{I}(b[\Gamma, x \in A])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(B[\Gamma, x \in A])$ in $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$.

The interpretation of the application $\mathsf{Ap}^{A,(x)B}(c,a)[\Gamma]$ is defined as the unique arrow directed towards the pullback of $\widetilde{\mathcal{I}(a[\Gamma])}$ along $\mathsf{pr}_{[\mathcal{I}(\Gamma), \mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A])]}$ (which is the middle rectangle in the following diagram defined as in lemma 4.20) making the following diagram in **Cont** commute (with the notation of lemma 4.40)

$$\begin{tikzcd}
{[\mathcal{I}(\Gamma), \mathbf{1}]} \arrow[rr, "\mathcal{I}(a[\Gamma]) \times \mathcal{I}(c[\Gamma])"] \arrow[dr, "\mathcal{I}(\mathsf{Ap}^{A,(x)B}(c,a)[\Gamma])"] \arrow[dddr, "id"'] & & {[\mathcal{I}(\Gamma), \mathcal{I}(A[\Gamma]) \times \Pi(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))]} \arrow[dr, "\mathsf{ev}_{\Pi} \circ \simeq_1"] & \\
& {[\mathcal{I}(\Gamma), \mathbf{Col}_{\widetilde{\mathcal{I}(a[\Gamma])}}(\mathcal{I}(B[\Gamma, x \in A]))]} \arrow[rr] \arrow[d, "\mathsf{pr}"] & & {[\mathcal{I}(\Gamma), \mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A])]} \arrow[d, "\mathsf{pr}"] \\
& \mathcal{I}(\Gamma) \arrow[rr, "\widetilde{\mathcal{I}(a[\Gamma])}"] \arrow[d, "\simeq_2"] & & {[\mathcal{I}(\Gamma), \mathcal{I}(A[\Gamma])]} \\
& {[\mathcal{I}(\Gamma), \mathbf{1}]} \arrow[urr, "\mathcal{I}(a[\Gamma])"'] & &
\end{tikzcd}$$

where $\simeq_1$ is the isomorphism from $[\mathcal{I}(\Gamma), \mathcal{I}(A[\Gamma]) \times \Pi(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))]$ to $[\mathcal{I}(\Gamma), \mathcal{I}(A[\Gamma]), \mathbf{Col}_{\mathbf{pr}_{[\Gamma, \mathcal{I}(A[\Gamma])]}}\Pi(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))]$ in **Cont** defined as in lemma 4.12 thanks to lemma 4.39 and $\simeq_2$ is the inverse of the isomorphism $\mathbf{pr}_{[\Gamma, \mathbf{1}]}$.

This arrow exists thanks to corollary 4.37, provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$, $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}([\Gamma]))$ and, finally, $\mathcal{I}(c[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\Pi(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$ in $\mathbf{Col}(\mathcal{I}([\Gamma]))$.

5.6 Disjoint sums

We interpret the disjoint sum as follows:

$$\mathcal{I}(A + B[\Gamma]) := \mathcal{I}(A[\Gamma]) +^{\mathcal{I}(\Gamma)} \mathcal{I}(B[\Gamma])$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the first injection of the disjoint sum $\mathsf{inl}^{A,B}(a)[\Gamma]$ is defined as the composed arrow making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "\mathcal{I}(a[\Gamma])"] \arrow[dr, swap, "\mathcal{I}(\mathsf{inl}^{A,B}(a)[\Gamma])"] & \mathcal{I}(A[\Gamma]) \arrow[d, "\mathsf{j}_1"] \\
& \mathcal{I}(A[\Gamma]) + \mathcal{I}(B[\Gamma])
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the second injection of the disjoint sum $\mathsf{inr}^{A,B}(b)[\Gamma]$ is defined as the composed arrow making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "\mathcal{I}(b[\Gamma])"] \arrow[dr, swap, "\mathcal{I}(\mathsf{inr}^{A,B}(b)[\Gamma])"] & \mathcal{I}(B[\Gamma]) \arrow[d, "\mathsf{j}_2"] \\
& \mathcal{I}(A[\Gamma]) + \mathcal{I}(B[\Gamma])
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(b[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(B[\Gamma])$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the eliminator of the disjoint sum $\mathsf{El}_{+}^{A,B,C}(c, (x)\,d, (y)\,e)[\Gamma]$ is defined as $f \circ \mathcal{I}(c[\Gamma])$ in the following commuting diagram in $\mathbf{Col}(\mathcal{I}(\Gamma))$ (with the notation of lemma 4.40)

$$
\begin{tikzcd}
 & \mathbf{1} \arrow[d, "\mathcal{I}(c[\Gamma])"] \arrow[dd, bend right, "\mathcal{I}(\mathsf{El}_{+}^{A,B,C}(c,(x)\,d,(y)\,e)[\Gamma])"'] & \\
\mathcal{I}(A[\Gamma]) \arrow[r, "\mathsf{j}_1"] \arrow[dr, "\mathcal{I}(d[\Gamma,x\in A])^{\mathcal{I}(\Gamma)}_{\natural}"'] & \mathcal{I}(A[\Gamma]) + \mathcal{I}(B[\Gamma]) \arrow[d, "f"] & \mathcal{I}(B[\Gamma]) \arrow[l, "\mathsf{j}_2"'] \arrow[dl, "\mathcal{I}(e[\Gamma,y\in B])^{\mathcal{I}(\Gamma)}_{\natural}"] \\
 & \mathcal{I}(C[\Gamma]) &
\end{tikzcd}
$$

where the existence and uniqueness of f is guaranteed by lemma 4.26, provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$, $\mathcal{I}(B[\Gamma])$ and $\mathcal{I}(C[\Gamma])$ are well defined objects of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(c[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma]) + \mathcal{I}(B[\Gamma])$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(d[\Gamma, x \in A])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{Col}_{\mathsf{pr}}(\mathcal{I}(C[\Gamma]))$ in $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$ and, finally, $\mathcal{I}(e[\Gamma, y \in B])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{Col}_{\mathsf{pr}}(\mathcal{I}(C[\Gamma]))$ in $\mathbf{Col}(\mathcal{I}([\Gamma, y \in B]))$.

5.7 Lists

We interpret the type of lists on a type as follows:

$$\mathcal{I}(\mathsf{List}(A)[\Gamma]) := \mathbf{List}^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]))$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the empty list $\epsilon^A[\Gamma]$ is defined as

$$\mathcal{I}(\epsilon^A[\Gamma]) := \epsilon : \mathbf{1} \to \mathbf{List}(\mathcal{I}(A[\Gamma]))$$

in $\mathbf{Col}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the list constructor $\mathsf{cons}^A(b, a)[\Gamma]$ is defined as the composed arrow making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$
\begin{tikzcd}
\mathbf{1} \arrow[r, "\langle \mathcal{I}(b[\Gamma]), \mathcal{I}(a[\Gamma]) \rangle"] \arrow[dr, "\mathcal{I}(\mathsf{cons}^A(b,a)[\Gamma])"'] & \mathbf{List}(\mathcal{I}(A[\Gamma])) \times \mathcal{I}(A[\Gamma]) \arrow[d, "\mathsf{cons}"] \\
 & \mathbf{List}(\mathcal{I}(A[\Gamma]))
\end{tikzcd}
$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(b[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{List}(\mathcal{I}(A[\Gamma]))$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$ while $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$.

The interpretation of the list eliminator $\mathsf{El}_{\mathsf{List}}^{A,B}(a, b, (x, y)\, c)[\Gamma]$ is defined as the composed arrow $f \circ \mathcal{I}(a[\Gamma])$ making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$ (with notation as in lemma 4.40)

$$\begin{tikzcd}
& \mathbf{1} \arrow[d, "\mathcal{I}(a[\Gamma])"'] \arrow[dd, bend left, "\mathcal{I}(\mathsf{El}_{\mathsf{List}}^{A,B}(a,b,(x,y)\,c)[\Gamma])"] & \\
\mathbf{1} \arrow[r, "\epsilon"'] \arrow[dr, "\mathcal{I}(b[\Gamma])"'] & \mathbf{List}(\mathcal{I}(A[\Gamma])) \arrow[d, "f"] & \mathbf{List}(\mathcal{I}(A[\Gamma])) \times \mathcal{I}(A[\Gamma]) \arrow[l, "\mathsf{cons}"'] \arrow[d, "f \times id"] \\
& \mathcal{I}(B[\Gamma]) & \mathcal{I}(B[\Gamma]) \times \mathcal{I}(A[\Gamma]) \arrow[l, "\mathcal{I}(c[\Gamma, x\in B, y\in A])^{\mathcal{I}(\Gamma)}_{\natural}"]
\end{tikzcd}$$

where the existence and uniqueness of f is guaranteed by lemma 4.28, provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{List}(\mathcal{I}(A[\Gamma]))$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(b[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(B[\Gamma])$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(c[\Gamma, x \in B, y \in A])$ is a well defined arrow from $\mathbf{1}$ to $\mathbf{Col}_{\mathsf{pr}^{(2)}}(\mathcal{I}(B[\Gamma]))$ in $\mathbf{Col}([\mathcal{I}(\Gamma), \mathcal{I}(B[\Gamma]), \mathbf{Col}_{\mathsf{pr}}(\mathcal{I}(A[\Gamma]))])$.

5.8 Collection of small propositions

The collection of small propositions is interpreted as follows:

$$\mathcal{I}(\mathsf{Prop}_s[\Gamma]) := \mathsf{USP}^{\mathcal{I}(\Gamma)}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**.

Recalling lemma 4.61, we define the interpretation of terms as follows.

The interpretation of the falsum code $\widehat{\bot}[\Gamma]$ is defined as

$$\mathcal{I}(\widehat{\bot}[\Gamma]) := \widehat{\bot} : \mathbf{1} \to \mathsf{USP}^{\mathcal{I}(\Gamma)}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**.

The interpretations of the conjunction code $a \mathbin{\widehat{\wedge}} b\,[\Gamma]$, the disjunction code $a \mathbin{\widehat{\vee}} b\,[\Gamma]$ and the implication code $a \mathbin{\widehat{\Rightarrow}} b\,[\Gamma]$ are defined as the composed arrows making the following diagrams commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd} \mathsf{USP}^{\mathcal{I}(\Gamma)} \times \mathsf{USP}^{\mathcal{I}(\Gamma)} \arrow[r, "\widehat{\wedge}"] & \mathsf{USP}^{\mathcal{I}(\Gamma)} \\ \mathbf{1} \arrow[u, "\langle \mathcal{I}(a[\Gamma]), \mathcal{I}(b[\Gamma]) \rangle"] \arrow[ur, swap, "\mathcal{I}(a \mathbin{\widehat{\wedge}} b[\Gamma])"] & \end{tikzcd} \qquad \begin{tikzcd} \mathsf{USP}^{\mathcal{I}(\Gamma)} \times \mathsf{USP}^{\mathcal{I}(\Gamma)} \arrow[r, "\widehat{\vee}"] & \mathsf{USP}^{\mathcal{I}(\Gamma)} \\ \mathbf{1} \arrow[u, "\langle \mathcal{I}(a[\Gamma]), \mathcal{I}(b[\Gamma]) \rangle"] \arrow[ur, swap, "\mathcal{I}(a \mathbin{\widehat{\vee}} b[\Gamma])"] & \end{tikzcd}$$

$$\begin{tikzcd} \mathsf{USP}^{\mathcal{I}(\Gamma)} \times \mathsf{USP}^{\mathcal{I}(\Gamma)} \arrow[r, "\widehat{\rightarrow}"] & \mathsf{USP}^{\mathcal{I}(\Gamma)} \\ \mathbf{1} \arrow[u, "\langle \mathcal{I}(a[\Gamma]), \mathcal{I}(b[\Gamma]) \rangle"] \arrow[ur, swap, "\mathcal{I}(a \mathbin{\widehat{\rightarrow}} b[\Gamma])"] & \end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(a[\Gamma])$ and $\mathcal{I}(b[\Gamma])$ are well defined arrows from $\mathbf{1}$ to $\mathsf{USP}^{\mathcal{I}(\Gamma)}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretations of the existential quantification code $(\widehat{\exists x \in A})b[\Gamma]$ and the universal quantification code $(\widehat{\forall x \in A})b[\Gamma]$ are defined as the composed arrows making the following diagrams commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd} \mathbf{1} \arrow[rr, "\mathbf{fam}_p(\mathcal{I}(b[\Gamma, x \in A]))"] \arrow[drr, swap, "\mathcal{I}((\widehat{\exists x \in A})b[\Gamma])"] & & \Sigma(\mathsf{US}^{\mathcal{I}(\Gamma)}, \tau^{\mathcal{I}(\Gamma)} \Rightarrow \mathsf{USP}^{[\mathcal{I}(\Gamma), \mathsf{US}^{\mathcal{I}(\Gamma)}]}) \arrow[d, "\widehat{\exists}"] \\ & & \mathsf{USP} \end{tikzcd}$$

$$\begin{tikzcd} \mathbf{1} \arrow[rr, "\mathbf{fam}_p(\mathcal{I}(b[\Gamma, x \in A]))"] \arrow[drr, swap, "\mathcal{I}((\widehat{\forall x \in A})b[\Gamma])"] & & \Sigma(\mathsf{US}^{\mathcal{I}(\Gamma)}, \tau^{\mathcal{I}(\Gamma)} \Rightarrow \mathsf{USP}^{[\mathcal{I}(\Gamma), \mathsf{US}^{\mathcal{I}(\Gamma)}]}) \arrow[d, "\widehat{\forall}"] \\ & & \mathsf{USP} \end{tikzcd}$$

where $\mathbf{fam}_p(\mathcal{I}(b[\Gamma, x \in A]))$ is defined using the notation in definition 4.66 and lemma 4.40 as

$$\langle \mathbf{en}_s^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma])), \mathsf{Cur}(\mathcal{I}(b[\Gamma, x \in A])_{\natural}^{\mathcal{I}(\Gamma)} \circ \pi_2^{\mathbf{1}, \mathcal{I}(A[\Gamma])}) \rangle_{\Sigma}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object in **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Set}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(b[\Gamma, x \in A])$ is a well defined arrow from $\mathbf{1}$ to $\mathsf{USP}^{\mathcal{I}([\Gamma, x \in A])}$ in $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$.

The interpretations of the propositional equality code $\widehat{\mathsf{Eq}}(A, a, b)[\Gamma]$ is defined as the composed arrow making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathbf{1} \arrow[rr, "\langle\ \mathbf{en}_s^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]))\ ,\ \langle \mathcal{I}(a[\Gamma]), \mathcal{I}(b[\Gamma])\rangle\ \rangle_\Sigma"] \arrow[drr, "\mathcal{I}(\widehat{\mathsf{Eq}}(A,a,b)[\Gamma])"'] & & \Sigma(\mathsf{US}^{\mathcal{I}(\Gamma)}, \tau^{\mathcal{I}(\Gamma)} \times \tau^{\mathcal{I}(\Gamma)}) \arrow[d, "\widehat{\mathsf{Eq}}"] \\
& & \mathsf{USP}^{\mathcal{I}(\Gamma)}
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is well defined object of $\mathbf{Set}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(a[\Gamma])$ and $\mathcal{I}(b[\Gamma])$ are well defined arrows from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

5.9 Collection of sets

We interpret the collection of sets as follows

$$\mathcal{I}(\mathsf{Set}[\Gamma]) := \mathsf{US}^{\mathcal{I}(\Gamma)}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**.

Recalling lemma 4.61, we define the interpretation of terms as follows.

The interpretation of the empty set code $\widehat{\mathsf{N}_0}$, the singleton code $\widehat{\mathsf{N}_1}$ and the natural numbers set code $\widehat{\mathsf{N}}$ are defined as follows: $\mathcal{I}(\widehat{\mathsf{N}_0}[\Gamma]) := \widehat{\mathsf{N}}_0 : \mathbf{1} \to \mathsf{US}^{\mathcal{I}(\Gamma)}$, $\mathcal{I}(\widehat{\mathsf{N}_1}[\Gamma]) := \widehat{\mathsf{N}}_1 : \mathbf{1} \to \mathsf{US}^{\mathcal{I}(\Gamma)}$ and $\mathcal{I}(\widehat{\mathsf{N}}[\Gamma]) := \widehat{\mathsf{N}} : \mathbf{1} \to \mathsf{US}^{\mathcal{I}(\Gamma)}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$, all provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**.

The interpretation of the disjoint sum code $a \mathbin{\widehat{+}} b[\Gamma]$ is defined as the composed arrow making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathsf{US}^{\mathcal{I}(\Gamma)} \times \mathsf{US}^{\mathcal{I}(\Gamma)} \arrow[r, "\widehat{+}"] & \mathsf{US}^{\mathcal{I}(\Gamma)} \\
\mathbf{1} \arrow[u, "\langle \mathcal{I}(a[\Gamma]), \mathcal{I}(b[\Gamma])\rangle"] \arrow[ur, "\mathcal{I}(a \mathbin{\widehat{+}} b[\Gamma])"']
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(a[\Gamma])$ and $\mathcal{I}(b[\Gamma])$ are well defined arrows from $\mathbf{1}$ to $\mathsf{US}^{\mathcal{I}(\Gamma)}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the list set code $\widehat{\mathsf{List}}(a)[\Gamma]$ is defined as the composed arrow making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathsf{US}^{\mathcal{I}(\Gamma)} \arrow[r, "\widehat{\mathsf{List}}"] & \mathsf{US}^{\mathcal{I}(\Gamma)} \\
\mathbf{1} \arrow[u, "\mathcal{I}(a[\Gamma])"] \arrow[ur, "\mathcal{I}(\widehat{\mathsf{List}}(a)[\Gamma])"']
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathsf{US}^{\mathcal{I}(\Gamma)}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the dependent sum code $(\widehat{\Sigma x \in A})b[\Gamma]$ and the dependent product code $(\widehat{\Pi x \in A})b[\Gamma]$ are defined as the composed arrows making the following diagrams commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "\mathbf{fam}(\mathcal{I}(b[\Gamma,x\in A]))"] \arrow[dr, "\mathcal{I}((\widehat{\Sigma x\in A})b[\Gamma])"'] & \Sigma(\mathsf{US}^{\mathcal{I}(\Gamma)}, \tau^{\mathcal{I}(\Gamma)} \Rightarrow \mathsf{US}^{[\mathcal{I}(\Gamma),\mathsf{US}^{\mathcal{I}(\Gamma)}]}) \arrow[d, "\widehat{\Sigma}"] \\
& \mathsf{US}^{\mathcal{I}(\Gamma)}
\end{tikzcd}$$

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "\mathbf{fam}(\mathcal{I}(b[\Gamma,x\in A]))"] \arrow[dr, "\mathcal{I}((\widehat{\Pi x\in A})b[\Gamma])"'] & \Sigma(\mathsf{US}^{\mathcal{I}(\Gamma)}, \tau^{\mathcal{I}(\Gamma)} \Rightarrow \mathsf{US}^{[\mathcal{I}(\Gamma),\mathsf{US}^{\mathcal{I}(\Gamma)}]}) \arrow[d, "\widehat{\Pi}"] \\
& \mathsf{US}^{\mathcal{I}(\Gamma)}
\end{tikzcd}$$

where $\mathbf{fam}(\mathcal{I}(b[\Gamma, x \in A]))$ is defined using the notation in definition 4.66 and lemma 4.40 as

$$\langle \mathbf{en}_s^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma])), \mathsf{Cur}(\mathcal{I}(b[\Gamma, x \in A])_\natural^{\mathcal{I}(\Gamma)} \circ \pi_2^{\mathbf{1},\mathcal{I}(A[\Gamma])})\rangle_\Sigma$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is well defined object of $\mathbf{Set}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(b[\Gamma, x \in A])$ is a well defined arrow from $\mathbf{1}$ to $\mathsf{US}^{\mathcal{I}([\Gamma,x\in A])}$ in $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$.

The interpretation of the small proposition code $\sigma(a)[\Gamma]$ is defined as the composed arrow making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "\mathcal{I}(a[\Gamma])"] \arrow[dr, "\mathcal{I}(\sigma(a)[\Gamma])"'] & \mathsf{USP}^{\mathcal{I}(\Gamma)} \arrow[d, "\sigma"] \\
& \mathsf{US}^{\mathcal{I}(\Gamma)}
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathsf{USP}^{\mathcal{I}(\Gamma)}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

5.10 Collection of propositional functions

We interpret the collection of propositional functions as follows:

$$\mathcal{I}(A \Rightarrow \mathsf{Prop}_s[\Gamma]) := \mathcal{I}(A[\Gamma]) \Rightarrow^{\mathcal{I}(\Gamma)} \mathsf{USP}^{\mathcal{I}(\Gamma)}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the propositional function lambda-abstraction $(\lambda x)^{A}_{\Rightarrow} b[\Gamma]$ is defined as the unique arrow (see lemma 4.27) making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$ (with notation in lemma 4.40)

$$\begin{tikzcd}
\mathcal{I}(A[\Gamma]) \arrow[rr, "\mathcal{I}(b[\Gamma,x\in A])^{\mathcal{I}(\Gamma)}_{\natural}"] \arrow[d, "\langle !,id\rangle"] & & \mathsf{USP}^{\mathcal{I}(\Gamma)} \\
\mathbf{1}\times\mathcal{I}(A[\Gamma]) \arrow[rr, "\mathcal{I}((\lambda x)^{A}_{\Rightarrow} b[\Gamma])\times id"] & & (\mathcal{I}(A[\Gamma]) \Rightarrow \mathsf{USP}^{\mathcal{I}(\Gamma)})\times\mathcal{I}(A[\Gamma]) \arrow[u, "\mathsf{ev}"']
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(b[\Gamma, x \in A])$ is a well defined arrow from $\mathbf{1}$ to $\mathsf{USP}^{\mathcal{I}([\Gamma,x\in A])}$ in $\mathbf{Col}(\mathcal{I}([\Gamma, x \in A]))$.

The interpretation of the propositional function application $\mathsf{Ap}^{A}_{\Rightarrow}(c, a)[\Gamma]$ is defined as the composed arrow making the following diagram commute in $\mathbf{Col}(\mathcal{I}(\Gamma))$

$$\begin{tikzcd}
\mathbf{1} \arrow[r, "\langle\mathcal{I}(c[\Gamma]),\mathcal{I}(a[\Gamma])\rangle"] \arrow[dr, "\mathcal{I}(\mathsf{Ap}^{A}_{\Rightarrow}(c,a)[\Gamma])"'] & (\mathcal{I}(A[\Gamma]) \Rightarrow \mathsf{USP}^{\mathcal{I}(\Gamma)})\times\mathcal{I}(A[\Gamma]) \arrow[d, "\mathsf{ev}"] \\
& \mathsf{USP}^{\mathcal{I}(\Gamma)}
\end{tikzcd}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(c[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma]) \Rightarrow \mathsf{USP}^{\mathcal{I}(\Gamma)}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

5.11 Falsum

We interpret falsum as follows:

$$\mathcal{I}(\bot[\Gamma]) := \bot^{\mathcal{I}(\Gamma)}$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**.

The interpretation of the falsum eliminator $\mathsf{r}_0^{A}(a)[\Gamma]$ is given by

$$\mathcal{I}(\mathsf{r}_0^{A}(a)[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma])$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object in **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Prop}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(a[\Gamma]) : \top \sqsubseteq \bot$ is well defined in $\mathbf{Prop}(\mathcal{I}(\Gamma))$.

5.12 Conjunction

We interpret the conjunction as follows:

$$\mathcal{I}(A \wedge B[\Gamma]) := \mathcal{I}(A[\Gamma]) \sqcap_{\mathcal{I}(\Gamma)} \mathcal{I}(B[\Gamma])$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Prop}(\mathcal{I}(\Gamma))$.

The interpretation of the conjunction pairing $\langle a, b\rangle_{\wedge}^{A,B}[\Gamma]$ is defined as

$$\mathcal{I}(\langle a, b\rangle_{\wedge}^{A,B}[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma]) \sqcap \mathcal{I}(B[\Gamma])$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Prop}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(a[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma])$ and $\mathcal{I}(b[\Gamma]) : \top \sqsubseteq \mathcal{I}(B[\Gamma])$ are well defined in $\mathbf{Prop}(\mathcal{I}(\Gamma))$.

The interpretations of the conjunction projections $\pi_{\wedge,1}^{A,B}(c)[\Gamma]$ and $\pi_{\wedge,2}^{A,B}(c)[\Gamma]$ are defined as

$$\mathcal{I}(\pi_{\wedge,1}^{A,B}(c)[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma]) \qquad \mathcal{I}(\pi_{\wedge,2}^{A,B}(c)[\Gamma]) : \top \sqsubseteq \mathcal{I}(B[\Gamma])$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Prop}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(c[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma]) \sqcap \mathcal{I}(B[\Gamma])$ is well defined in $\mathbf{Prop}(\mathcal{I}(\Gamma))$.

5.13 Disjunction

We interpret the disjunction as follows:

$$\mathcal{I}(A \vee B[\Gamma]) := \mathcal{I}(A[\Gamma]) \sqcup_{\mathcal{I}(\Gamma)} \mathcal{I}(B[\Gamma])$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Prop}(\mathcal{I}(\Gamma))$.

The interpretations of disjunction injections $\mathsf{inl}_{\vee}^{A,B}(a)[\Gamma]$ and $\mathsf{inr}_{\vee}^{A,B}(b)[\Gamma]$ are defined as

$$\mathcal{I}(\mathsf{inl}_{\vee}^{A,B}(a)[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma]) \sqcup \mathcal{I}(B[\Gamma]) \qquad \mathcal{I}(\mathsf{inr}_{\vee}^{A,B}(b)[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma]) \sqcup \mathcal{I}(B[\Gamma])$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are both well defined objects of $\mathbf{Prop}(\mathcal{I}(\Gamma))$ and finally, when interpreting the first injection $\mathcal{I}(a[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma])$ is also assumed to be well defined in $\mathbf{Prop}(\mathcal{I}(\Gamma))$, and when interpreting the second injection $\mathcal{I}(b[\Gamma]) : \top \sqsubseteq \mathcal{I}(B[\Gamma])$ is also assumed to be well defined in $\mathbf{Prop}(\mathcal{I}(\Gamma))$.

The interpretation of the disjunction eliminator $\mathsf{El}_{\vee}^{A,B,C}(c, (x)\, d, (y)\, e)[\Gamma]$ is defined as

$$\mathcal{I}(\mathsf{El}_{\vee}^{A,B,C}(c, (x)\, d, (y)\, e)[\Gamma]) : \top \sqsubseteq \mathcal{I}(C[\Gamma])$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of $\mathbf{Cont}$, $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Prop}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(c[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma]) \sqcup \mathcal{I}(B[\Gamma])$ is well defined in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(d[\Gamma, x \in A]) : \top \sqsubseteq \mathbf{Prop}_{\mathsf{pr}}(\mathcal{I}(C[\Gamma]))$ is well defined in $\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A]))$ and $\mathcal{I}(e[\Gamma, y \in B]) : \top \sqsubseteq \mathbf{Prop}_{\mathsf{pr}}(\mathcal{I}(C[\Gamma]))$ is well defined in $\mathbf{Prop}(\mathcal{I}([\Gamma, y \in B]))$.

5.14 Implication

We interpret implication as follows:

$$\mathcal{I}(A \to B[\Gamma]) := \mathcal{I}(A[\Gamma]) \to_{\mathcal{I}(\Gamma)} \mathcal{I}(B[\Gamma])$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of $\mathbf{Cont}$ and $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Prop}(\mathcal{I}(\Gamma))$.

The interpretation of the implication lambda-abstraction $(\lambda x)_{\to}^{A,B}(b)[\Gamma]$ is defined as

$$\mathcal{I}((\lambda x)_{\to}^{A,B}(b)[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma]) \to \mathcal{I}(B[\Gamma])$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of $\mathbf{Cont}$, $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Prop}(\mathcal{I}(\Gamma))$ and

$$\mathcal{I}(b[\Gamma, x \in A]) : \top \sqsubseteq \mathbf{Prop}_{\mathsf{pr}}(\mathcal{I}(B[\Gamma]))$$

is well defined in $\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A]))$.

The interpretation of the implication application $\mathsf{Ap}_{\to}^{A,B}(c, a)[\Gamma]$ is defined as

$$\mathcal{I}(\mathsf{Ap}_{\to}^{A,B}(c, a)[\Gamma]) : \top \sqsubseteq \mathcal{I}(B[\Gamma])$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of $\mathbf{Cont}$, $\mathcal{I}(A[\Gamma])$ and $\mathcal{I}(B[\Gamma])$ are well defined objects of $\mathbf{Prop}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(c[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma]) \to \mathcal{I}(B[\Gamma])$ and $\mathcal{I}(a[\Gamma]) : \top \sqsubseteq \mathcal{I}(A[\Gamma])$ are well defined in $\mathbf{Prop}(\mathcal{I}([\Gamma]))$.

5.15 Existential quantifier

We interpret the existential quantifier as follows:

$$\mathcal{I}((\exists x \in A)B[\Gamma]) := \exists^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A]))$.

The interpretation of the existential quantifier pairing $\langle a, b\rangle_{\exists}^{A,(x)B}[\Gamma]$ is defined as

$$\mathcal{I}(\langle a, b\rangle_{\exists}^{A,(x)B}[\Gamma]) : \top^{\mathcal{I}(\Gamma)} \sqsubseteq_{\mathcal{I}(\Gamma)} \exists^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined objects of $\mathbf{Col}(\Gamma)$ and $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A]))$ and furthermore, $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}([\Gamma]))$ and $\mathcal{I}(b[\Gamma]) : \top \sqsubseteq \mathbf{Prop}_{\widetilde{\mathcal{I}(a[\Gamma])}}(\mathcal{I}(B[\Gamma, x \in A]))$ is well defined in $\mathbf{Prop}(\mathcal{I}([\Gamma]))$ (see 4.32 for notation).

The interpretation of the existential quantifier eliminator $\mathsf{El}_{\exists}^{A,(x)B,C}(a, (x, y)\, b)[\Gamma]$ is defined as

$$\mathcal{I}(\mathsf{El}_{\exists}^{A,(x)B,C}(a, (x, y)\, b)[\Gamma]) : \top^{\mathcal{I}(\Gamma)} \sqsubseteq_{\mathcal{I}(\Gamma)} \mathcal{I}(C[\Gamma])$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\Gamma)$, $\mathcal{I}(C[\Gamma])$ is a well defined object of $\mathbf{Prop}(\Gamma)$ and $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A]))$ and furthermore

$$\mathcal{I}(a[\Gamma]) : \top \sqsubseteq \exists(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$$

is well defined in $\mathbf{Prop}(\mathcal{I}([\Gamma]))$ and

$$\mathcal{I}(b[\Gamma, x \in A, y \in B]) : \top \sqsubseteq \mathbf{Prop}_{\mathsf{pr}^{(2)}}(\mathcal{I}(C[\Gamma]))$$

is well defined in $\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A, y \in B]))$.

5.16 Universal quantifier

We interpret the universal quantifier as follows:

$$\mathcal{I}((\forall x \in A)B[\Gamma]) := \forall^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A]))$.

The interpretation of the universal quantifier lambda-abstraction $(\lambda x)_{\forall}^{A,B} b[\Gamma]$ is defined as

$$\mathcal{I}((\lambda x)_{\forall}^{A,B} b[\Gamma]) : \top^{\mathcal{I}(\Gamma)} \sqsubseteq_{\mathcal{I}(\Gamma)} \forall^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of

$\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A]))$ and $\mathcal{I}(b[\Gamma, x \in A]) : \top \sqsubseteq \mathcal{I}(B[\Gamma, x \in A])$ is well defined in $\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A]))$.

The interpretation of the universal quantifier application $\mathsf{Ap}_{\forall}^{A,(x)B}(c, a)[\Gamma]$ is defined in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ with the notation in 4.32 as

$$\mathcal{I}(\mathsf{Ap}_{\forall}^{A,(x)B}(c, a)[\Gamma]) : \top^{\mathcal{I}(\Gamma)} \sqsubseteq_{\mathcal{I}(\Gamma)} \mathbf{Prop}_{\widetilde{\mathcal{I}(a[\Gamma])}}(\mathcal{I}(B[\Gamma, x \in A]))$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$, $\mathcal{I}(B[\Gamma, x \in A])$ is a well defined object of $\mathbf{Prop}(\mathcal{I}([\Gamma, x \in A]))$ and, furthermore,

$$\mathcal{I}(c[\Gamma]) : \top \sqsubseteq \forall(\mathcal{I}(A[\Gamma]), \mathcal{I}(B[\Gamma, x \in A]))$$

is well defined in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(a[\Gamma]) : \mathbf{1} \to \mathcal{I}(A[\Gamma])$ is well defined in $\mathbf{Col}(\mathcal{I}([\Gamma]))$.

5.17 Equality proposition

We interpret the propositional equality as follows:

$$\mathcal{I}(\mathsf{Eq}(A, a, b)[\Gamma]) := \mathbf{Eq}^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(a[\Gamma]), \mathcal{I}(b[\Gamma]))$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and both $\mathcal{I}(a[\Gamma])$ and $\mathcal{I}(b[\Gamma])$ are well defined arrows from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

The interpretation of the propositional equality term $\mathsf{eq}_A(a)[\Gamma]$ is defined as

$$\mathcal{I}(\mathsf{eq}_A(a)[\Gamma]) : \top^{\mathcal{I}(\Gamma)} \sqsubseteq_{\mathcal{I}(\Gamma)} \mathbf{Eq}^{\mathcal{I}(\Gamma)}(\mathcal{I}(A[\Gamma]), \mathcal{I}(a[\Gamma]), \mathcal{I}(a[\Gamma]))$$

in $\mathbf{Prop}(\mathcal{I}(\Gamma))$ provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont**, $\mathcal{I}(A[\Gamma])$ is a well defined object of $\mathbf{Col}(\mathcal{I}(\Gamma))$ and $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathcal{I}(A[\Gamma])$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

5.18 Decoding

$$\mathcal{I}(\tau(a)[\Gamma]) := \tau_{sp}^{\mathcal{I}(\Gamma)}(\mathcal{I}(a[\Gamma]))$$

provided that $\mathcal{I}(\Gamma)$ is a well defined object of **Cont** and $\mathcal{I}(a[\Gamma])$ is a well defined arrow from $\mathbf{1}$ to $\mathsf{USP}^{\mathcal{I}(\Gamma)}$ in $\mathbf{Col}(\mathcal{I}(\Gamma))$.

6 Validity theorem

We start with defining a list of arrows useful to interpret telescopic substitutions of dependent type theory in realized contexts Γ of $\widehat{ID_1}$.

Definition 6.1. *Suppose* Γ *and* Γ' *are objects of* $\mathbf{Cont}$*. We define simultaneously by induction on the length of the realized context* Γ

1. *a list of arrows* $\overline{a} = [a_1, ..., a_{\ell(\Gamma)}]$ *in* $\mathbf{Col}(\Gamma')$ *with domain* $\mathbf{1}^{\Gamma'}$ *called* instance of substitution *for* Γ *in context* Γ'

2. *an arrow* $\mathbf{sub}(\overline{a},\, \Gamma',\, \Gamma) : \Gamma' \to \Gamma$ *for every instance of substitution* $\overline{a}$

as follows:

1. *the empty list* $[\,]$ *is an instance of substitution for* $[\,]$ *in context* Γ' *and* $\mathbf{sub}([\,],\, \Gamma',\, [\,]) :=!_{\Gamma',[\,]} : \Gamma' \to [\,]$

2. *if* $[\Gamma, B]$ *is an object of* $\mathbf{Cont}$*, then* $[\overline{a}, b]$ *is an instance of substitution for* $[\Gamma, B]$ *in context* Γ' *if and only if* $\overline{a}$ *is an instance of substitution for* Γ *in context* Γ' *and* b *is an arrow from* $\mathbf{1}$ *to* $\mathbf{Col}_{\mathbf{sub}(\overline{a},\, \Gamma',\, \Gamma)}(B)$ *in* $\mathbf{Col}(\Gamma')$*.*

   In this case $\mathbf{sub}([\overline{a}, b],\, \Gamma',\, [\Gamma, B])$ *is defined as* $q(\mathbf{sub}(\overline{a},\, \Gamma',\, \Gamma),\, [\Gamma, B]) \circ \widetilde{b}$ *with the notation in 4.32:*

$$\begin{tikzcd}
\Gamma' \arrow[d, "\widetilde{b}"'] \arrow[drr, "{\mathbf{sub}([\overline{a},b],\Gamma',[\Gamma,B])}"] & & \\
{[\Gamma', \mathbf{Col}_{\mathbf{sub}(\overline{a},\, \Gamma',\, \Gamma)}(B)]} \arrow[rr, "{q(\mathbf{sub}(\overline{a},\Gamma',\Gamma),[\Gamma,B])}"'] \arrow[d, "\mathsf{pr}"] & & {[\Gamma, B]} \arrow[d, "\mathsf{pr}"] \\
\Gamma' \arrow[rr, "{\mathbf{sub}(\overline{a},\Gamma'\,\Gamma)}"'] & & \Gamma
\end{tikzcd}$$

Remark 6.2. *Notice that there is a bijection between lists of arrows which are instances of substitution for* Γ *in context* Γ' *and arrows in* $\mathbf{Cont}$ *from* Γ' *to* Γ*.*

The following lemma, which can be proved by induction on the definition of the syntax in precontext, shows that weakening is interpreted as one could expect.

Lemma 6.3 (weakening)**.** *Suppose* $[\Gamma, \Gamma', \Gamma'']$ *is a precontext such that both* $[\Gamma, \Gamma']$ *and* $[\Gamma, \Gamma'']$ *are precontexts. Suppose that the length of* Γ*,* Γ' *and* Γ'' *are* n, n', n'' *respectively and that* $\mathcal{I}([\Gamma, \Gamma'])$ *and* $\mathcal{I}([\Gamma, \Gamma''])$ *are well defined. Then* $\mathcal{I}([\Gamma, \Gamma', \Gamma''])$ *is well defined and if* $[\Gamma, \Gamma', \Gamma'']$ *is* $[y_1 \in A_1, ..., y_{n+n'+n''} \in A_{n+n'+n''}]$*, then*

1. *the list* $\overline{\mathbf{weak}}$ *defined by*

$$[\mathcal{I}(y_1[\Gamma,\Gamma',\Gamma'']),...,\mathcal{I}(y_n[\Gamma,\Gamma',\Gamma'']),\mathcal{I}(y_{n+n'+1}[\Gamma,\Gamma',\Gamma'']),...,\mathcal{I}(y_{n+n'+n''}[\Gamma,\Gamma',\Gamma''])]$$

   is an instance of substitution for $\mathcal{I}([\Gamma,\Gamma''])$ *in context* $\mathcal{I}([\Gamma,\Gamma',\Gamma''])$

2. *if* $\mathcal{I}(A[\Gamma,\Gamma''])$ *is well defined, then* $\mathcal{I}(A[\Gamma,\Gamma',\Gamma''])$ *is well defined and it coincides with*

$$\mathbf{Col}_{\mathbf{sub}(\overline{\mathbf{weak}},\mathcal{I}([\Gamma,\Gamma',\Gamma'']),\mathcal{I}([\Gamma,\Gamma'']))}(\mathcal{I}(A[\Gamma,\Gamma'']))$$

3. *if* $\mathcal{I}(a[\Gamma,\Gamma''])$ *is well defined, then* $\mathcal{I}(a[\Gamma,\Gamma',\Gamma''])$ *is well defined and it coincides with*

$$\mathbf{Col}_{\mathbf{sub}(\overline{\mathbf{weak}},\mathcal{I}([\Gamma,\Gamma',\Gamma'']),\mathcal{I}([\Gamma,\Gamma'']))}(\mathcal{I}(a[\Gamma,\Gamma'']))$$

The next lemma can be proved by induction on the definition of the syntax in precontext and it shows that substitution commutes with the interpretation $\mathcal{I}$, i. e. that one can first perform a substitution in $\mathbf{mTT}^a$ and then interpret the resulting type or term or, equivalently, first interpret terms and types of $\mathbf{mTT}^a$ and then perform the substitution of the interpreted terms.

Lemma 6.4 (Substitution Lemma)**.** *Let* $\Gamma = [x_1 \in A_1, ..., x_n \in A_n]$ *be a precontext with* $n > 0$ *and let* Γ' *be a precontext. Let* $\mathcal{I}(\Gamma)$ *and* $\mathcal{I}(\Gamma')$ *be well defined and suppose that* $\mathcal{I}(a_1[\Gamma'])$*,....,* $\mathcal{I}(a_n[\Gamma'])$ *are well defined and constitute an instance of subtitution for* $\mathcal{I}(\Gamma)$ *in context* $\mathcal{I}(\Gamma')$*. Then*

1. *if* $\mathcal{I}(B[\Gamma])$ *is well defined in* $\mathbf{Col}(\mathcal{I}(\Gamma))$*, then* $\mathcal{I}(B[a_1/x_1,...,a_n/x_n][\Gamma'])$ *is well defined and it coincides with*

$$\mathbf{Col}_{\mathbf{sub}(\mathcal{I}(\overline{a}[\Gamma']),\mathcal{I}(\Gamma'),\mathcal{I}(\Gamma))}(\mathcal{I}(B[\Gamma]))$$

2. *if* $\mathcal{I}(b[\Gamma])$ *is well defined in* $\mathbf{Col}(\mathcal{I}(\Gamma))$*, then* $\mathcal{I}(b[a_1/x_1,...,a_n/x_n][\Gamma'])$ *is well defined and it coincides with*

$$\mathbf{Col}_{\mathbf{sub}(\mathcal{I}(\overline{a}[\Gamma']),\mathcal{I}(\Gamma'),\mathcal{I}(\Gamma))}(\mathcal{I}(b[\Gamma]))$$

where we denote by $\mathcal{I}(\overline{a}[\Gamma'])$ *the list of the interpretations* $\mathcal{I}(a_i[\Gamma'])$*.*

What shown so far helps to prove our main theorem:

Theorem 6.5. *The effective pretripos* $(\mathbf{Cont},\mathbf{Col},\mathbf{Set},\mathbf{Prop},\mathbf{Prop}_s)$ *validates all judgements of* $\mathbf{mTT}^a$ *in the sense that:*
for every judgement J *of* $\mathbf{mTT}^a$*, if* $\mathbf{mTT}^a \vdash J$*, then* $\mathcal{R} \vDash J$*.*

Thanks to proposition 2.1 we also deduce

Corollary 6.6. *The effective pretripos* $(\mathbf{Cont}, \mathbf{Col}, \mathbf{Set}, \mathbf{Prop}, \mathbf{Prop}_s)$ *validates all judgements of* $\mathbf{mTT}$ *in the sense that:*

for every judgement J *of* $\mathbf{mTT}$*, if* $\mathbf{mTT} \vdash J$*, then* $\mathcal{R} \vDash J$.

6.1 The validity of CT

Proposition 6.7. *The effective pretripos validates* $\mathbf{CT}$*, i.e.* $\mathcal{R} \vDash \mathbf{CT}$*, where* $\mathbf{CT}$ *is the formula*

$$(\forall x \in \mathsf{N})\,(\exists y \in \mathsf{N})\,R(x,y) \rightarrow (\exists e \in \mathsf{N})\,(\forall x \in \mathsf{N})\,(\exists z \in \mathsf{N})\,(T(e,x,z) \wedge R(x, U(z)))$$

where T *and* U *are respectively the Kleene predicate and the primitive recursive function representing Kleene application in* $\mathbf{mTT}^a$ *respectively.*

Proof. The validity in $\mathcal{R}$ of $\mathbf{CT}$ can be obtained as a consequence of the validity in $\mathcal{R}$ of the following principles:
Formal Church thesis for type-theoretic functions $\mathbf{CT}_\lambda$ defined as:

$$\begin{aligned}(\forall f \in (\Pi x \in \mathsf{N})\mathsf{N})\,(\exists e \in \mathsf{N})\,(\forall x \in \mathsf{N})\,(\exists y \in \mathsf{N})&\\ (T(e,x,y)\ \wedge\ \mathsf{Eq}(\mathsf{N}, U(y), \mathsf{Ap}(f,x)))&\end{aligned}$$

and *the axiom of countable choice* $\mathbf{AC}_{\mathsf{N},\mathsf{N}}$ defined for $\mathbf{mTT}^a \vdash R\,prop\,[x \in \mathsf{N}, y \in \mathsf{N}]$ as

$$(\forall x \in \mathsf{N})\,(\exists y \in \mathsf{N})\,R(x,y) \rightarrow (\exists f \in (\Pi x \in \mathsf{N})\,\mathsf{N})\,(\forall x \in \mathsf{N})\,R(x, \mathsf{Ap}(f,x))$$

One can easily show that in $\mathbf{Prop}([\,])$:

$$\mathbf{1} \sqsubseteq \mathcal{I}(\mathbf{CT}_\lambda[\,]).$$

In fact we know by general results on Kleene realizability that there exists a numeral $\mathbf{r}$ for which

$$HA\ \vdash\ \exists u\, T(f,x,u) \rightarrow (\{\mathbf{r}\}(f,x)\ \Vdash_k\ \exists u\, T(f,x,u))$$

Using this remark and proof irrelevance we can show that the interpretation of $\mathbf{CT}_\lambda[\,]$ has a global element determined by the numeral

$$\Lambda z.\Lambda f.\,\{\mathbf{p}\}\,(f,\ \Lambda x.\,\{\mathbf{p}\}\,(\{\mathbf{p}_1\}\,(\{\mathbf{r}\}(f,x)),\ \{\mathbf{p}\}(\{\mathbf{p}_2\}\,(\{\mathbf{r}\}\,(f,x)),\ 0)))$$

where the first variable z belongs to $\mathbf{1}$. Moreover $\mathcal{R} \vDash \mathbf{AC}_{\mathsf{N},\mathsf{N}}$ as equality in N is interpreted as numerical equality. $\square$

It is worth noting that theorem 6.6 shows that $\widehat{ID_1}$ is an upper bound of the proof-theoretic strength of **mTT**. Actually, this is a direct proof of it because in [23] it was observed that **mTT** can be interpreted in first-order Martin-Löf's type theory with one universe, for short MLtt_1, whose proof-theoretic strength is known to be equal to that of $\widehat{ID_1}$(see [8]). Even more our interpretation of **mTT** in $\widehat{ID_1}$ is a modification of that in [8] used to establish that $\widehat{ID_1}$ is an upper bound of MLtt_1. The main difference between our proof and that for MLtt_1 is that ours validates **CT** while that in [8] falsifies **CT**.

It is left to future work to establish whether the proof-theoretic strength of **mTT** and hence of **MF** coincides with that of $\widehat{ID_1}$ as it happens to MLtt_1.

7 Conclusions

We have built here an effective predicative categorical structure, called *effective pretripos* for the intensional level **mTT** of **MF** extended with the formal Church thesis **CT**, in Feferman's predicative classical theory $\widehat{ID_1}$.

This is intended to be a basic categorical structure of realizers for **mTT** useful to build a predicative variant of Hyland's Effective Topos. A predicative effective topos will be obtained by completing our effective pretripos with quotients by means of the elementary quotient completion introduced and studied in [26, 25, 27]. Indeed, such an elementary quotient completion axiomatizes the quotient model used in [23] to interpret the extensional level of **MF** into **mTT** and generalizes the notion of the exact completion on a lex category. Therefore, it appears to be a starting point to generalize the tripos-to-topos construction in [18] predicatively and to validate the extensional level **emTT** of **MF** extended with **CT** when applied to our effective pretripos.

Another goal of our future work will be to make a precise comparison between our categorical structures of realizers for **MF** and the categorical approach to predicative effective models for Aczel's CZF in [41].

Acknowledgements. We acknowledge many useful fruitful discussions with Laura Crosilla, Takako Nemoto, Giovanni Sambin and Thomas Streicher on topics of this paper. We are grateful to Andrew Swan very much for proof-reading our paper and to the referees for their suggestions.

References

[1] P. Aczel. The type theoretic interpretation of constructive set theory. In *Logic Colloquium '77 (Proc. Conf., Wrocław, 1977)*, volume 96 of *Stud. Logic Foundations Math.* North-Holland, Amsterdam-New York, 1978.

[2] P. Aczel. The type theoretic interpretation of constructive set theory: choice principles. In Dirk van Dalen Anne Troelstra, editor, *The L.E.J. Brouwer Centenary Symposium (Noordwijkerhout, 1981)*, volume 110 of *Stud. Logic Foundations Math.* North-Holland, Amsterdam-New York, 1982.

[3] P. Aczel. The type theoretic interpretation of constructive set theory: inductive definitions. In *Logic, methodology and philosophy of science, VII (Salzburg, 1983)*, volume 114 of *Stud. Logic Foundations Math.* North-Holland, Amsterdam-New York, 1986.

[4] P. Aczel and M. Rathjen. Notes on constructive set theory. Mittag-Leffler Technical Report No.40, 2001.

[5] A. Asperti, W. Ricciotti, C. Sacerdoti Coen, and E. Tassi. The Matita interactive theorem prover. In *Proceedings of the 23rd International Conference on Automated Deduction (CADE-2011), Wroclaw, Poland*, volume 6803 of *LNCS*, 2011.

[6] Andrea Asperti, Wilmer Ricciotti, and Claudio Sacerdoti Coen. Matita tutorial. *Journal of Formalized Reasoning*, 7(2):91–199, 2014.

[7] G. Barthes, V. Capretta, and O. Pons. Setoids in type theory. *J. Funct. Programming*, 13(2):261–293, 2003. Special issue on "Logical frameworks and metalanguages".

[8] M. Beeson. *Foundations of Constructive Mathematics.* Springer-Verlag, Berlin, 1985.

[9] Y. Bertot and P. Castéran. *Interactive Theorem Proving and Program Development.* Texts in Theoretical Computer Science. Springer Verlag, 2004. ISBN-3-540-20854-2.

[10] A. Bove, P. Dybjer, and U. Norell. A brief overview of Agda - a functional language with dependent types. In S. Berghofer, T. Nipkow, C. Urban, and M. Wenzel, editors, *Theorem Proving in Higher Order Logics, 22nd International Conference, TPHOLs 2009*, volume 5674 of *Lecture Notes in Computer Science*, pages 73–78. Springer, August 2009.

[11] W. Bucholz, S. Feferman, W. Pohlers, and W. Sieg. *Iterated inductive definitions and subsystems of analysis.* Number 897 in Lecture Notes in Mathematics. Springer Verlag, 1981.

[12] Coq development team. *The Coq Proof Assistant Reference Manual: release 8.4pl6.* INRIA, Orsay, France, April 2015.

[13] T. Coquand. Metamathematical investigation of a calculus of constructions. In P. Odifreddi, editor, *Logic in Computer Science*, pages 91–122. Academic Press, 1990.

[14] T. Coquand and C. Paulin-Mohring. Inductively defined types. In P. Martin-Löf and G. Mints, editors, *Proceedings of the International Conference on Computer Logic (Colog '88)*, volume 417 of *Lecture Notes in Computer Science*, pages 50–66. Springer, Berlin, Germany, 1990.

[15] S. Feferman. Iterated inductive fixed-point theories: application to Hancock's conjecture. In *Patras Logic Symposion*, pages 171–196. North Holland, 1982.

[16] E. Griffor and M. Rathjen. The strenght of some martin löf type theories. *Archive for Mathematical Logic*, 33(5):347–385, 1994.

[17] J. M. E. Hyland. The effective topos. In A. S. Troesltra and D. van Dalen, editors, *The L. E. J. Brouwer centenary symposium*, Studies in logic and the foundations of

mathematics, pages 165–216. North-Holland, 1982.

[18] J. M. E. Hyland, P. T. Johnstone, and A. M. Pitts. Tripos theory. *Bull. Austral. Math. Soc.*, 88:205–232, 1980.

[19] A. Joyal and I. Moerdijk. *Algebraic set theory*, volume 220 of *Lecture Note Series*. Cambridge University Press, 1995.

[20] S. Mac Lane. *Categories for the working mathematician.*, volume 5 of *Graduate text in Mathematics*. Springer, 1971.

[21] S. Mac Lane and I. Moerdijk. *Sheaves in Geometry and Logic. A first introduction to Topos theory.* Springer Verlag, 1992.

[22] M. E. Maietti. Modular correspondence between dependent type theories and categories including pretopoi and topoi. *Mathematical Structures in Computer Science*, 15(6): 1089–1149, 2005.

[23] M. E. Maietti. A minimalist two-level foundation for constructive mathematics. *Annals of Pure and Applied Logic*, 160(3):319–354, 2009.

[24] M. E. Maietti and S. Maschio. An extensional Kleene realizability model for the Minimalist Foundation. In *20th International Conference on Types for Proofs and Programs, TYPES 2014, May 12-15, 2014, Paris, France*, pages 162–186, 2014.

[25] M. E. Maietti and G. Rosolini. Elementary quotient completion. *Theory and Applications of Categories*, 27(17):445–463, 2013.

[26] M. E. Maietti and G. Rosolini. Quotient completion for the foundation of constructive mathematics. *Logica Universalis*, 7(3):371–402, 2013.

[27] M. E. Maietti and G. Rosolini. Unifying exact completions. *Applied Categorical Structures*, 23(1):43–52, 2015.

[28] M. E. Maietti and G. Sambin. Toward a minimalist foundation for constructive mathematics. In L. Crosilla and P. Schuster, editor, *From Sets and Types to Topology and Analysis: Practicable Foundations for Constructive Mathematics*, number 48 in Oxford Logic Guides, pages 91–114. Oxford University Press, 2005.

[29] M. E. Maietti and G. Sambin. Why topology in the Minimalist Foundation must be pointfree. *Logic and Logical Philosophy*, 22(2):167–199, 2013.

[30] P. Martin-Löf. *Intuitionistic Type Theory. Notes by G. Sambin of a series of lectures given in Padua, June 1980.* Bibliopolis, Naples, 1984.

[31] B. Nordström, K. Petersson, and J. Smith. *Programming in Martin Löf's Type Theory.* Clarendon Press, Oxford, 1990.

[32] A. M. Pitts. Categorical logic. In Oxford University Press, editor, *Handbook of Logic in Computer Science*, volume 5, pages 39–128, 2000.

[33] A. M. Pitts. Tripos theory in retrospect. *Mathematical Structures in Computer Science*, 12:265–279, 2002.

[34] G. Sambin. Two applications of dynamic constructivism: Brouwer's continuity principle and choice sequences in formal topology. In M. van Atten, P. Boldini, M. Bourdeau, and G. Heinzmann, editors, *One Hundred years of Intuitionism (1907-2007). The Cerisy Conference*, pages 301–315. Birkhäuser, 2008.

[35] G. Sambin and S. Valentini. Building up a toolbox for Martin-Löf's type theory: subset theory. In G. Sambin and J. Smith, editors, *Twenty-five years of constructive type theory, Proceedings of a Congress held in Venice, October 1995*, pages 221–244. Oxford U. P., 1998.

[36] R. Seely. Locally cartesian closed categories and type theory. *Math. Proc. Cambr. Phyl. Soc.*, 95:33–48, 1984.

[37] R. A. G. Seely. Hyperdoctrines, natural deduction and the Beck condition. *Zeitschr. f. Math. Logik. und Grundlagen d. Math.*, 29:505–542, 1983.

[38] T. Streicher. *Semantics of type theory.* Birkhäuser, 1991.

[39] T. Streicher. Independence of the induction principle and the axiom of choice in the pure calculus of constructions. *Theoretical Computer Science*, 103(2):395–408, 1992.

[40] A. S. Troelstra and D. van Dalen. Constructivism in mathematics, an introduction, vol. I. In *Studies in logic and the foundations of mathematics.* North-Holland, 1988.

[41] B. van den Berg and I. Moerdijk. Aspects of predicative algebraic set theory II: Realizability. *Theoretical Computer Science*, Theoretical Computer 412:1916–1940, 2011.

Received \jreceived