

C-system of a module over a monad on sets¹

Keywords: Contextual categories, Lawvere theories, monads, modules, HOL, Mizar, Coq, Martin-Lof, derivable judgements.

Vladimir Voevodsky^{2,3} September 2015

Abstract

This is the second paper in a series started in [?] which aims to provide mathematical descriptions of objects and constructions related to the semantical theory of dependent type systems.

We construct for any pair $(\mathbf{R}, \mathbf{LM})$, where R is a monad on sets and LM is a left module over R , a C-system (“contextual category”) $CC(\mathbf{R}, \mathbf{LM})$ and describe, using the results of [?] a class of sub-quotients of $CC(\mathbf{R}, \mathbf{LM})$ in terms of objects directly constructed from R and LM . In the special case of the monads of expressions associated with binding signatures this construction gives, for the first time, a mathematically rigorous way of constructing a C-system from a general collection of judgements of the four Martin-Löf kinds that satisfies a well specified set of conditions.

Contents

1	Introduction	1
2	Pre-algebraic theories	7
3	Free pre-algebraic theories	12
4	The C-system $C(\mathbf{R})$	12
5	The C-system $CC(\mathbf{R}, \mathbf{LM})$	25
6	C-subsystems of $CC(\mathbf{R}, \mathbf{LM})$	37
7	Operations σ and $\tilde{\sigma}$	40
8	Regular sub-quotients of $CC(\mathbf{R}, \mathbf{LM})$	40

1 Introduction

The first few steps in all approaches to the semantics of dependent type theories remain insufficiently understood. The constructions which have been worked out in detail in the case of a few particular type systems by dedicated authors are being extended to the wide variety of type systems under consideration today by analogy. This is not acceptable in mathematics. Instead we should be able to obtain the required results for new type systems by *specialization* of general theorems formulated and proved for abstract objects the instances of which combine together to produce the objects associated with a given type system.

¹2000 *Mathematical Subject Classification*: 18D99, 08C99, 03B15 03F50,

²School of Mathematics, Institute for Advanced Study, Princeton NJ, USA. e-mail: vladimir@ias.edu

³Work on this paper was supported by NSF grant 1100938.

An approach that follows this general philosophy was outlined in [?]. In this approach the connection between the type theories, which belong to the concrete world of logic and programming, and abstract mathematical concepts such as sets or homotopy types is constructed through the intermediary of C-systems.

C-systems were introduced in [?] (see also [?]) under the name “contextual categories”. A modified axiomatics of C-systems and the construction of new C-systems as sub-objects and regular quotients of the existing ones in a way convenient for use in type-theoretic applications are considered in [?].

In the approach of [?], in order to provide a mathematical interpretation (semantics) for a type theory one constructs two C-systems. One C-system is constructed from the formulas of the type theory using as an initial step the construction of the present paper. The second C-system is constructed from the category of abstract mathematical objects using the results of [?]. Both C-systems are then equipped with additional operations corresponding to the “inference rules” of the type theory.

The main component of this approach is the expected result that for a particular class of the inference rules the concrete C-systems built using the constructions of the present paper and equipped with operations corresponding to these inference rules are initial objects in the category of C-systems with the corresponding operations. This is known as the Initiality Conjecture. In the case of the pure Calculus of Constructions this conjecture was proved in 1988 by Thomas Streicher [?]. The problem of finding an appropriate formulation of the general version of the conjecture and of proving this general version will be the subject of future work.

For such inference rules, then, there are unique homomorphisms from the concrete C-systems to the abstract C-systems that are compatible with the corresponding systems of operations. Since objects and morphisms of concrete C-systems are built from formulas of the type theory and objects and morphisms of abstract C-systems are built from mathematical objects such as sets or homotopy types and the corresponding functions, these homomorphisms provide a mathematical meaning to formulas of type theory.

The existence of such homomorphisms in the particular case of the “standard univalent models” of Martin-Löf type theories and of the Calculus of Inductive Constructions (CIC) provides the only known justification for the use of the proof assistants such as Coq for the formalization of mathematics in the univalent style (see [?], [?]).

It is important to distinguish the concepts of a model of a type theory and the concept of an interpretation of the same type theory. A *model* of type theory can be defined as a C-system that is equipped with the systems of operations corresponding to the inference rules of the type theory. A (categorical) *interpretation* of a type theory with values in a given category \mathcal{C} is a functor from the category underlying the syntactic C-system of the type theory to \mathcal{C} .

Only if we know that the initiality result holds for a given type theory can we claim that any its model defines an interpretation by taking the composition of the canonical homomorphism of the C-systems with the functor such as the functor *int* of [?]. A similar problem also arises in the predicate logic but there, since one considers only one fixed system of syntax and inference rules, it can and had been solved once without the development of a general theory.

When one speaks about the univalent model of a Martin-Löf type theory or of the CIC one often fails to distinguish between the model and the interpretation. A construction of a *model* for the version of the Martin-Löf type theory that is used in the UniMath library ([?],[?]) was sketched in [?]. At the time when that paper was written it was unfortunately assumed that a proof of

the initiality result can be found in the existing body of work on type theory which is reflected in [?, Theorem 1.2.9] (cf. also [?, Example 1.2.3] that claims as obvious everything that is done in both the present paper and in [?]). Since then it became clear that this is not the case and that a mathematical theory leading to the initiality theorem and providing a proof of such a theorem is lacking and needs to be developed.

As the criteria for what constitutes an acceptable proof were becoming more clear as a result of continuing work on formalization, it also became clear that more detailed and general proofs need to be given to many of the theorems of [?] that are related to the model itself. For the two of the several main groups of inference rules of current type theories it is done in [?] and [?]. Other groups of inference rules will be considered in further papers of the series.

This paper may be considered to be an analog of [?] for the concrete side of the theory in the sense that it provides a very general construction the particular cases of which lead to the concrete (syntactic) C-systems of type theories.

If $R = (R, \eta, \mu)$ is a monad on a category \mathcal{C} (see Definition ??) then there is defined the Kleisli category \mathcal{C}_R of R whose objects are the same as objects of \mathcal{C} and morphisms from X to Y are defined as morphisms from X to $R(Y)$ in \mathcal{C} . The identity morphisms in \mathcal{C}_R are given by the η operation of R and the composition by the composition in \mathcal{C} and the μ operation of R .

A left R -module LM over R with values in a category \mathcal{D} (see Definition ??) defines a functor $LM_R : \mathcal{C}_R \rightarrow \mathcal{D}$ and this function from left R -modules to functors from the Kleisli category is an equivalence (see ??).

An important case is the left R -module corresponding to R itself which we will also denote by R .

Monads on the category of sets and left modules over such monads have a number of special ????

Of a particular interest is the case of “syntactic” pairs $(\mathbf{R}, \mathbf{LM})$ where for $X = \{x_1, \dots, x_n\}$, $R(X)$ and $LM(X)$ are the sets of expressions of some kind with free variables from $\{x_1, \dots, x_n\}$ modulo an equivalence relation such as α -equivalence. The difference between R and LM is in this case expressed by the fact that one can substitute elements from $R(X)$ for variables both in $R(Y)$ and $LM(Y)$ but elements of $LM(X)$ can not be substituted for variables in either.

The simplest class of syntactic pairs, where $LM = R$, arises from binding signatures (see [?, p.228]). To any such signature Σ one associates a class of expressions with bindings and $R(\{x_1, \dots, x_n\})$ is the set of such expressions with free variables from the set $\{x_1, \dots, x_n\}$ modulo α -equivalence.

The more general case when LM is not equal to R arises when one starts to distinguish “type expressions” from “object expressions”. The rules of type theories require the possibility to substitute an object expression instead of a variable both in a type expression and in an object expression but do not require to substitute a type expression instead of a variable either in a type or in an object expression. In type theories of proof assistants such as Coq the user may be under the impression that the substitution of type expressions instead of variables occurs (as in substituting *unit* for T in *iscontr*(T) in the UniMath to obtain *iscontr*(*unit*), cf. [?]) this is however due to a “silent” map from object expressions to type expressions that is used in these theories. What actually happens in these substitutions is that an object expression whose type is a universe is substituted instead of a variable in some situations and the same object expression is mapped to the set of type expressions and used as a type expression in others. In our constructions this corresponds to $LM = R$ - an object expression that is an element of $R(X)$ for some set of variables X is considered as an element of the set of type expressions $LM(X)$ using the identity map defined by this equality (more generally one may observe the same illusion when $LM \subset R$).

The question of whether to keep this map silent or to give it a name (usually El) is known in type theory as the difference between the type theories with “Russell universes” (silent map) and “Tarski universes” (explicit map) which is at the center of some of the current controversies about the universe management in proof assistants. It is also the subject of a discussion in the last, unfinished, chapter in [?].

For the purposes of the present paper we fortunately don’t need to make a choice between the two approaches since the formalism that we develop is applicable to both. It is however clear from the constructions that the separation between R and LM is a very natural possibility that directly generalizes the case of $LM \subset R$ and creates new examples (e.g. Example ??).

As was shown in [?] the monad that one associates to a binding signature can be characterized as being an initial object in the category of monads equipped with “left-linear” operations corresponding to the operations of the signature. This provides an abstract mathematical characterization of the concrete objects - expressions modulo α -equivalence or, equivalently, expressions with De Bruijn indexes.

An important remark needs to be made here. While monads provide a very convenient way of expressing syntax with bindings in terms familiar to mathematicians the approach based on monads is equivalent to an earlier one pioneered in [?]. For two sets X and Y let $Fun(X, Y)$ be the set of functions from X to Y . In that earlier approach one considers the category F such that $Ob(F) = \mathbf{N}$ and

$$Mor(F) = \coprod_{m,n} Fun(stn(m), stn(n))$$

where $stn(i) = \{0, \dots, i - 1\}$ is the “standard” set with i elements, and functors $Funct(F, Sets)$ from F to $Sets$ (the authors call these functors “presheaves” considering them as presheaves on F^{op}). This category of functors is equivalent⁴ to the category of finitary (co-continuous) functors from $Sets$ to $Sets$. In particular, there is a monoidal structure (\bullet, V) on $Funct(F, Sets)$ corresponding to the composition of functors under this equivalence (cf. [?, Sec. 3]) and finitary monads can be considered as monoids in $Funct(F, Sets)$ with respect to this monoidal structure.

Using this equivalence of concepts (detailed in []) the constructions and results of [?] and [?] can be viewed together as describing different aspects of a fundamental connection between the concrete world of syntax and the abstract world of categorical mathematics.

After this long detour let me clarify that the results and constructions of the present paper do not depend on either [?] or [?], except for the definition of a left module over a monad in [?] and examples. The connection to [?] and [?] will become important only in future papers where we will consider the abstract concept of a system of inference rules and where binding signatures and the corresponding syntactic monads will become essential.

In the present paper, after some general comments about monads on $Sets$ and their modules, we construct for any such monad R and a left module LM over R a C-system (contextual category) $CC(\mathbf{R}, \mathbf{LM})$. We start with a construction of a category $C(\mathbf{R})$ such that $Ob(C(\mathbf{R})) = \mathbf{N}$ is the set of natural numbers whose elements we will denote as \hat{m}, \hat{n} etc. and

$$Mor(C(\mathbf{R})) = \coprod_{\hat{m}, \hat{n}} Hom_{Sets_R}(stn(n), stn(m))$$

⁴In the set-theoretic mathematics this equivalence can not be defined without axiom of choice. The problem lies in the fact that the obvious functor from F to the category of finite sets, while it is fully faithful and essentially surjective, does not have a constructive inverse. In the univalent foundations, while one still can not construct an inverse to the functor from F to finite sets, one can construct an inverse to the corresponding functor from $Funct(FSets, Sets)$ to $Funct(F, Sets)$ using the fact that $Sets$ is a (univalent) category. Cf. [?] and [?, RezkCompletion library].

and the identity and composition is defined such as to make the mapping $\widehat{n} \mapsto \text{stn}(n)$ to extend to a fully faithful functor cr_1 from $C(\mathbf{R})^{op}$ to the Kleisli category Sets_R of R . We may sometimes use this functor as a “coercion”, in the terminology of proof assistant Coq, i.e., to write \widehat{n} instead of $\text{stn}(n)$ and f instead of $\Phi(f)$. We will also use the function $LM \mapsto LM_R$ from left modules to functors on the Kleisli category as a coercion. In agreement with this convention we may write LM for the presheaf of sets on $C(\mathbf{R})$ given by $\widehat{n} \mapsto LM(n)$.

We describe, using the results of [?], all the C-subsystems of $CC(\mathbf{R}, \mathbf{LM})$ in terms of objects directly associated with R and LM .

We then define two additional operations σ and $\tilde{\sigma}$ on $CC(\mathbf{R}, \mathbf{LM})$ and describe the regular congruence relations (see [?]) on C-subsystems of $CC(\mathbf{R}, \mathbf{LM})$ which are compatible in a certain sense with σ and $\tilde{\sigma}$.

Such regular congruence relations correspond, in the particular cases of syntactic monads and C-subsystems of $CC(\mathbf{R}, \mathbf{LM})$ generated by systems of inference rules, to the relations that can be described by the two kinds of equality judgements.

More precisely, suppose that we are given a type theory that is formulated in terms of the four kinds of judgements originally introduced by Per Martin-Löf in [?, p.161]⁵:

$$\begin{aligned} & (x_0 : T_0, \dots, x_{n-1} : T_{n-1}) T \text{ type} \\ & (x_0 : T_0, \dots, x_{n-1} : T_{n-1}) t : T \\ & (x_0 : T_0, \dots, x_{n-1} : T_{n-1}) T = T' \\ & (x_0 : T_0, \dots, x_{n-1} : T_{n-1}) t = t' : T \end{aligned}$$

to which one adds the judgement

$$(x_0 : T_0, \dots, x_{n-1} : T_{n-1}) ok$$

asserting that $(x_0 : T_0, \dots, x_{n-1} : T_{n-1})$ is a valid context of variable declarations.

Since we are only interested in the α -equivalence classes of judgements we may assume that the variables declared in the context are taken from the set of natural numbers such that the first declared variable is 0, the second is 1 etc. Then, the set of judgements of the form

$$(0 : T_0, \dots, n-1 : T_{n-1}) T \text{ type}$$

can be identified with the set of judgements of the form

$$(0 : T_0, \dots, n-1 : T_{n-1}, n : T) ok$$

With this identification the derivable judgements of the type theory whose raw syntax for object expressions is given by a monad R and raw syntax for type expressions by a left R -module LM , can be described as four subsets $\widetilde{B}, B, \text{Beq}$ and $\widetilde{\text{Beq}}$ where

$$\widetilde{B} \subset \prod_{n \in \mathbf{N}} LM(0) \times \dots \times LM(n-1)$$

⁵We are not using the notation based on \triangleright that became widespread in the modern literature on type theory since it conflicts with other uses of the turnstile symbol in logic.

$$\begin{aligned}
B &\subset \prod_{n \in \mathbf{N}} LM(0) \times \dots \times LM(n-1) \times R(stn(n)) \times LM(n) \\
Beq &\subset \prod_{n \in \mathbf{N}} LM(0) \times \dots \times LM(n-1) \times LM(n)^2 \\
\widetilde{Beq} &\subset \prod_{n \in \mathbf{N}} LM(0) \times \dots \times LM(n-1) \times R(stn(n))^2 \times LM(n)
\end{aligned}$$

The sets on the right hand side of the first two of these inclusions are in the bijective correspondences with the sets $Ob(CC(\mathbf{R}, \mathbf{LM}))$ and $\widetilde{Ob}(CC(\mathbf{R}, \mathbf{LM}))$. It was shown in [?, Proposition 4.3] that for any C-system CC , pairs (B, \widetilde{B}) where $B \subset Ob(CC)$ and $\widetilde{B} \subset \widetilde{Ob}(CC)$ that satisfy certain conditions are in a bijective correspondence with C-subsystems of CC . In Proposition 6.1 we give a direct reformulation of these conditions in the case of C-systems of the form $CC(\mathbf{R}, \mathbf{LM})$ in terms of subsets \widetilde{B} and B and in Remark 6.2 we show how these conditions look like in the notation of type theory.

We then continue our analysis to provide a mathematical meaning to the subsets Beq and \widetilde{Beq} as well. In order to obtain a bijection of Proposition 8.11 between pairs of such subsets that satisfy certain properties and objects that have meaning for general C-systems we introduce operations σ and $\widetilde{\sigma}$.

Proposition 8.2 and subsequent lemmas culminating in Proposition 8.11 form what is probably the most important part of the paper. They provide, for the first time, a rigorous mathematical analysis of the conditions that the derivable definitional equality judgements of a type system have to satisfy in order to define well-behaved equivalence relation on the sets such as the sets of morphisms (context substitutions) of a type theory.

While proving conditions of Proposition 8.2 in the case when \widetilde{B} , B , Beq and \widetilde{Beq} are the sets of derivable judgements of a particular type system is something that must be done in order to apply the results of the present paper to this type system, proving these conditions is much less difficult than giving a direct construction of a C-system starting from the syntax and the inference rules.

Providing this explicit set of conditions and proving that they are necessary and sufficient in order to associate a C-system and, therefore, any of the other semantic objects such as a category with families, to a particular type system may be considered to be the main result of this paper.

The paper is written in the formalization-ready style with the intent to produce a UniMath formalization of its results in the near future.

We also consider it to be important to formalize the results of this paper in HOL or Zermelo-Fraenkel theory. Indeed, since it forms a part of the theory on which the relative consistency of the UniMath language with respect to the set theory is based, it has to be formally verified in a theory that is weaker than UniMath or, better, that is weaker than both UniMath and the Zermelo-Fraenkel theory. However, the main choices that we had to make were made with the intent to first formalize this paper in the UniMath.

For morphisms $f : X \rightarrow Y$ and $g : Y \rightarrow Z$ we denote their composition as $f \circ g$. For functors $F : \mathcal{C} \rightarrow \mathcal{C}'$, $G : \mathcal{C}' \rightarrow \mathcal{C}''$ we use the standard notation $G \circ F$ for their composition.

Following the notation of the proof assistant Coq we let *unit* denote the distinguished one point set or type and *tt* the only point of *unit*.

This is one of the papers extending the material which I started to work on in [?]. I would like to thank the Institute Henri Poincaré in Paris and the organizers of the “Proofs” trimester for their hospitality during the preparation of this paper. The work on this paper was facilitated by discussions with Richard Garner and Egbert Rijke.

2 Pre-algebraic theories

For two sets X and Y we let $Fun(X, Y)$ denote the set of functions from X to Y . For $n \in \mathbf{N}$ we let $stn(n)$ denote the “standard” set of n elements $\{0, \dots, n-1\}$.

Definition 2.1 [2015.08.17.def1] *A pre-algebraic theory is a collection of data of the following form*

1. a function $R : \mathbf{N} \rightarrow Sets$,
2. for each n a function $\eta_n \in Fun(stn(n), R(n))$,
3. for each $f \in Fun(stn(m), R(n))$ a function $bind(f) \in Fun(R(m), R(n))$,

such that the following conditions hold:

1. for all n , $bind(\eta_n) = Id_{R(n)}$,
2. for all $f \in Fun(stn(m), stn(n))$, $\eta_n \circ bind(f) = f$,
3. for all $f \in Fun(stn(k), R(m))$, $g \in Fun(stn(m), R(n))$, $bind(f \circ bind(g)) = bind(f) \circ bind(g)$.

Remark 2.2 [2015.08.19.rem2] The type of pre-algebraic theories is equivalent to the type of clones as defined in [?, Section 3]. Whether this equivalence is an isomorphism in the syntactic category depends on how one formalizes expressions such as X^n that are used in [?] but it is clear that these definitions are very close to each other.

Let us introduce the following notation:

$$F(m, n) = Fun(stn(m), stn(n))$$

and, for a pre-algebraic theory R ,

$$R(m, n) = Fun(stn(m), R(n))$$

Let $\mathbf{R} = (R, \eta, bind)$ be a pre-algebraic theory. For each pair $m, n \in \mathbf{N}$ define a function

$$\phi_{\mathbf{R}} : F(m, n) \rightarrow R(m, n)$$

by the formula

$$\phi_{\mathbf{R}}(f) = f \circ \eta_m$$

For each l, m, n and $f \in R(l, m)$, $g \in R(m, n)$ define $f \hat{\circ} g \in R(l, n)$ by the formula

$$[2015.08.26.eq8] f \hat{\circ} g = f \circ bind(g) \tag{1}$$

Lemma 2.3 [2015.08.18.11] *Let $\mathbf{R} = (R, \eta, \text{bind})$ be a pre-algebraic theory. Then one has:*

1. *for any k, l, m, n and $f \in R(k, l)$, $g \in R(l, m)$, $h \in R(m, n)$ one has*

$$(f \hat{\circ} g) \hat{\circ} h = f \hat{\circ} (g \hat{\circ} h)$$

2. *for any $f \in R(m, n)$ one has*

$$f \hat{\circ} \eta_n = f$$

$$\eta_m \hat{\circ} f = f$$

3. *for any $f \in F(l, m)$, $g \in R(m, n)$ one has*

$$\phi_{\mathbf{R}}(f) \hat{\circ} g = f \circ g$$

4. *for any $f \in F(l, m)$, $g \in F(m, n)$ one has*

$$\phi_{\mathbf{R}}(f) \hat{\circ} \phi_{\mathbf{R}}(g) = \phi_{\mathbf{R}}(f \circ g)$$

Proof:

1. We have

$$(f \hat{\circ} g) \hat{\circ} h = (f \circ \text{bind}(g)) \circ \text{bind}(h) = f \circ (\text{bind}(g) \circ \text{bind}(h)) = f \circ \text{bind}(g \circ \text{bind}(h)) = f \hat{\circ} (g \hat{\circ} h)$$

2. We have

$$f \hat{\circ} \eta_n = f \circ \text{bind}(\eta_n) = f \circ \text{Id}_{R(n)} = f$$

$$\eta_m \hat{\circ} f = \eta_m \circ \text{bind}(f) = f$$

3. We have

$$\phi_{\mathbf{R}}(f) \hat{\circ} g = \phi_{\mathbf{R}}(f) \circ \text{bind}(g) = f \circ \eta_m \circ \text{bind}(g) = f \circ (\eta_m \circ \text{bind}(g)) = f \circ g$$

4. We have

$$\phi_{\mathbf{R}}(f) \hat{\circ} \phi_{\mathbf{R}}(g) = f \circ \phi_{\mathbf{R}}(g) = f \circ g \circ \eta_m = (f \circ g) \circ \eta_m = \phi_{\mathbf{R}}(f \circ g)$$

We are going to use the functions $\phi_{\mathbf{R}}$ as coercions in the terminology of the proof assistant Coq, i.e. when an element f of $F(m, n)$ occurs in a position where an element of $R(m, n)$ should be it is assumed that f has to be replaced by $\phi_{\mathbf{R}}(f)$ before the computation can occur. We will also use bind as a coercion so that when an element g of $R(m, n)$ occurs in a position where a function from $R(m)$ to $R(n)$ is expected it has to be replaced by $\text{bind}(g)$ before the computation can occur. In practice, since many of our lemmas assert equalities between expressions where these coercions are not inserted, we will be using expressions without coercions inserted or with coercions only partially inserted in our computations but all the computations and proofs would remain valid if the coercions were fully inserted.

Remark 2.4 [2015.08.22.rem2] We could also use the functions η_m as coercions. This however would lead to an ambiguity of the following form. For $f \in R(m, n)$ and $i \in stn(m)$ we could read $f(i)$ both directly as the result of application of $f \in Fun(stn(m), R(n))$ to i and as $bind(f)(\eta_m(i))$. While the corresponding elements in $R(n)$ are equal they are equal because of the equality $\eta_m \circ bind(f) = f$ from Definition 2.1. In the UniMath and other similar formalizations this equality is not a substitutional equality but only a transportational one and having the same expression to denote two objects that are not substitutionally equal is an ambiguity. See also Remark 2.7 below.

Since we will have to deal with elements of the sets of functions $R(m, n) = R(stn(n))^{stn(m)}$ and of similar sets such as the sets $Ob_n(CC(\mathbf{R}, \mathbf{LM}))$ introduced later we need to choose some way to represent them. For the purpose of the present paper we will write such elements as sequences, i.e., to denote the function, which in the notation of λ -calculus is written as $\lambda i : stn(n), f_i$, we will write (f_0, \dots, f_{n-1}) .

Lemma 2.5 [2015.08.26.13] *Let $f \in R(l, m)$ and $g \in R(m, n)$ then for $i = 0, \dots, n - 1$ one has*

$$[2015.08.26.eq7](f \hat{\circ} g)(i) = g(f(i)) \quad (2)$$

or, in the sequence notation

$$[2015.08.24.eq1](f_0, \dots, f_{m-1}) \hat{\circ} g = (g(f_0), \dots, g(f_{m-1})) \quad (3)$$

Proof: After we insert the coercions the equality (2) becomes

$$(f \hat{\circ} g)(i) = bind(g)(f(i))$$

which is equivalent to the definition (1) of $f \hat{\circ} g$.

Lemma 2.6 [2015.08.30.11] *Let $f \in F(l, m)$, $g \in R(m, n)$ and $i \in stn(l)$. Then one has*

$$[2015.08.26.eq4](f \hat{\circ} g)(i) = g(f(i)) \quad (4)$$

Proof: Inserting coercions we get the goal

$$(\phi_{\mathbf{R}}(f) \hat{\circ} g)(i) = g(f(i))$$

which follows by application of Lemma 2.3(3).

Remark 2.7 [2015.08.30.rem3] In the UniMath, or any similar, formalization there is a difference between the equalities of Lemmas 2.5 and 2.6. The equality of the first lemma is obtained by unfolding the definitions of $\hat{\circ}$ and of \circ . As such it is a substitutional equality which can be used to directly substitute the left hand side of (2) for the right hand side and vice versa. The equality of the second lemma is obtained by unfolding definitions and then applying the equality of Definition 2.1. As a declared equality it is only a transportational one which makes it impossible to use it to directly substitute one hand side of the equality for another. Instead a special “transport” function has to be used.

In this paper we will not distinguish between the two types of equality in part because this distinction is non-existent in either the Zermelo-Fraenkel theory or HOL. Nevertheless, some of our choices, such as the one explained in Remark 2.4, have been affected by the existence of this distinction.

For $n \in \mathbf{N}$ and $i = 0, \dots, n-1$ let

$$x_i^n = \eta_n(i) \in R(n)$$

then for $f \in R(n, m)$ we have

$$[\mathbf{2015.08.24.eq2}] f(x_i^n) = \text{bind}(f)(\eta_n(i)) = f(i) \quad (5)$$

In our sequence notations, for an element x of a set X , the expression (x) denotes the function $\text{stn}(1) \rightarrow X$ that takes 0 to x . With this fact in mind observe also that for $f \in R(n, m)$ one has

$$[\mathbf{2015.08.24.eq5}] (x_i^n) \hat{\circ} f = f(x_i^n) = f(i) \quad (6)$$

where the first equality is by (3) and the second one by (5).

Let

$$\partial_n^i : \text{stn}(n) \rightarrow \text{stn}(n+1)$$

for $0 \leq i \leq n$ be the increasing inclusion that does not take the value i and

$$\sigma_n^i : \text{stn}(n+2) \rightarrow \text{stn}(n+1)$$

for $0 \leq i \leq n$ be the increasing surjection that takes the value i twice. Taking into account that, in the notation of [?], $[n] = \text{stn}(n+1)$ these are the standard generators of the simplicial category Δ together with $\partial_0^0 : \text{stn}(0) \rightarrow \text{stn}(1)$.

In our sequence notation we have

$$[\mathbf{2015.08.24.eq7}] \phi_{\mathbf{R}}(\partial_n^i) = (x_0^{n+1}, \dots, x_{i-1}^{n+1}, x_{i+1}^{n+1}, \dots, x_n^{n+1}) \quad (7)$$

and

$$[\mathbf{2015.08.24.eq8}] \phi_{\mathbf{R}}(\sigma_n^i) = (x_0^{n+1}, \dots, x_{i-1}^{n+1}, x_i^{n+1}, x_i^{n+1}, x_{i+1}^{n+1}, \dots, x_n^{n+1}) \quad (8)$$

in particular

$$[\mathbf{2015.07.12.eq5}] \phi_{\mathbf{R}}(\partial_n^n) = (x_0^{n+1}, \dots, x_{n-1}^{n+1}) \quad (9)$$

Let

$$\iota_n^i : \text{stn}(n) \rightarrow \text{stn}(n+i)$$

be the function given by $\iota_n^i(j) = j$ for $j = 0, \dots, n-1$. Note that we have

$$[\mathbf{2015.08.22.eq7}] \iota_n^1 = \partial_n^n \quad (10)$$

and

$$\begin{aligned} [\mathbf{2015.08.22.eq8}] \iota_n^i(x_j^n) &= \text{bind}(\phi_{\mathbf{R}}(\iota_n^i))(\eta_n(j)) = (\eta_n \circ \text{bind}(\phi_{\mathbf{R}}(\iota_n^i)))(j) = \\ & \phi_{\mathbf{R}}(\iota_n^i)(j) = \eta_{n+i}(\iota_n^i(j)) = \eta_{n+i}(j) = x_j^{n+i} \end{aligned} \quad (11)$$

Lemma 2.8 *[2015.08.26.11] Let $f \in R(n+1, n)$. Then $\iota_n^1 \hat{\circ} f = \eta_n$ if and only if $f(i) = x_n^i$ for $i = 0, \dots, n-1$.*

Proof: For $i = 0, \dots, n-1$ we have

$$x_n^i = \eta_n(i) = (\iota_n^1 \widehat{\circ} f)(i) = f(\iota_n^1(i)) = f(i)$$

For $f \in R(n, m)$, $f = (f_0, \dots, f_{n-1})$ define an element $qq(f) \in R(n+1, m+1)$ by the formula:

$$\text{[2015.08.26.eq9]} qq(f) = (\iota_m^1(f_0), \dots, \iota_m^1(f_{n-1}), x_m^{m+1}) \quad (12)$$

Lemma 2.9 [2015.08.26.12] For $i \in \mathbf{N}$ and $f = (f_0, \dots, f_{n-1})$ in $R(n, m)$ one has

$$qq^i(f) = (\iota_m^i(f_0), \dots, \iota_m^i(f_{n-1}), x_m^{m+i}, \dots, x_{m+i-1}^{m+i})$$

Proof: Straightforward by induction on i .

Lemma 2.10 [2015.08.26.13a] For $n, i \in \mathbf{N}$ one has

$$qq^i(\iota_n^1) = \partial_{n+i}^n$$

Proof: Inserting coercions we see that we have to prove that

$$qq^i(\phi_{\mathbf{R}}(\iota_n^1)) = \phi_{\mathbf{R}}(\partial_n^{n+i})$$

We have $\phi_{\mathbf{R}}(\iota_n^1) = (x_0^{n+1}, \dots, x_{n-1}^{n+1})$. By Lemma 2.9 and (11) we get

$$\begin{aligned} qq^i(\phi_{\mathbf{R}}(\iota_n^1)) &= (\iota_{n+1}^i(x_0^{n+1}), \dots, \iota_{n+1}^i(x_{n-1}^{n+1}), x_{n+1}^{n+1+i}, \dots, x_{n+i}^{n+1+i}) = \\ &= (x_0^{n+1+i}, \dots, x_{n-1}^{n+1+i}, x_{n+1}^{n+1+i}, \dots, x_{n+i}^{n+1+i}) = \phi_{\mathbf{R}}(\partial_n^{n+i}) \end{aligned}$$

where the last equality is (7).

Lemma 2.11 [2015.08.28.11] For $i, m \in \mathbf{N}$ and $r \in R(m)$ one has

$$qq^i(x_0^m, \dots, x_{m-1}^m, r) = (x_0^{m+i}, \dots, x_{m-1}^{m+i}, \iota_m^i(r), x_m^{m+i}, \dots, x_{m+i-1}^{m+i})$$

Proof: One has

$$\begin{aligned} qq^i(x_0^m, \dots, x_{m-1}^m, r) &= (\iota_m^i(x_0^m), \dots, \iota_m^i(x_{m-1}^m), \iota_m^i(r), x_m^{m+i}, \dots, x_{m+i-1}^{m+i}) = \\ &= (x_0^{m+i}, \dots, x_{m-1}^{m+i}, \iota_m^i(r), x_m^{m+i}, \dots, x_{m+i-1}^{m+i}) = \end{aligned}$$

where the first equality is by Lemma 2.9 and the second one by (11).

Define, for any pre-algebraic theory \mathbf{R} , an operation $\sigma_{\mathbf{R}}$ such that form $m, n \in \mathbf{N}$, $n > m$ and $r \in R(m)$, $s \in R(n)$ one has

$$\text{[2015.09.07.eq1]} \sigma_{\mathbf{R}}(m, r, n, s) = qq^{n-m-1}(x_0^m, \dots, x_{m-1}^m, r)(s) =$$

$$(x_0^{n-1}, \dots, x_{m-1}^{n-1}, \iota_m^{n-m-1}(r), x_m^{n-1}, \dots, x_{n-2}^{n-1})(s) \quad (13)$$

This operation is a *dependent* function of the form

$$\sigma_{\mathbf{R}} : \{m \in \mathbf{N}, r \in R(m), n \in \mathbf{N}, s \in R(n) \mid n > m\} \rightarrow R(n-1),$$

i.e., a function whose codomain depends on the argument.

In the Zermelo-Fraenkel theory this can be expressed as follows. Let us fix a universe UU . Then a dependent function \mathbf{f} with a domain dom is a pair of functions $codom : dom \rightarrow UU$ and $f : dom \rightarrow UU$ such that for all $x \in dom$ one has $f(x) \in codom(x)$. Introducing a third parameter U instead of a fixed universe UU and taking equivalence classes with respect to the obvious relation obtained from inclusions $U \subset U'$ one can define dependent functions in set theory without using a universe.

In the language of UniMath and other similar “dependently typed” languages there is a syntactic mechanism for working with variable codomains of functions which are called type families. In such languages $\sigma_{\mathbf{R}}$ can be directly defined as dependent function with the codomain family given by $codom(\sigma_{\mathbf{R}})(m, r, n, s) = R(n-1)$.

Of the four arguments m, r, n and E of $\sigma_{\mathbf{R}}$ we will use m and n as implicit arguments, that is, arguments that are not written explicitly because they often can be inferred from the remaining arguments by a simple algorithmic procedure. In cases when m and n can not be inferred from r and E , which we will not encounter in this paper, the syntax of the proof assistant Coq is to write $@\sigma_{\mathbf{R}}$ instead of $\sigma_{\mathbf{R}}$ and to supply all four arguments explicitly.

3 Free pre-algebraic theories

The goal of this section is to show how to assign to any concrete type theory a pre-algebraic theory and how to characterize this pre-algebraic theory by its universal properties.

The approach that we use is essentially a re-writing of some of the results of [?] and [?] in terms of pre-algebraic theories.

The construction first assigns to a concrete type theory a binding signature. A binding signature, a concept introduced by Peter Aczel in [?], is a finite set of data - a sequence of sequences of natural numbers. We will show which sequences correspond to the pre-algebraic theories corresponding to such type theories as the Martin-Lof type theories MLTT72 and MLTT79.

The other is to characterize these pre-algebraic theories by their universal properties.

???

4 The C-system $C(\mathbf{R})$

Define a category $C(\mathbf{R})$ as follows. $Ob(C(\mathbf{R})) = \mathbf{N}$ is the set of natural numbers. The set of morphisms of $C(\mathbf{R})$ is given by

$$Mor(C(\mathbf{R})) = \amalg_{m,n \in \mathbf{N}} R(n, m)$$

with the obvious domain and codomain functions. To distinguish the positions in formulas where natural numbers are used as objects of $C(\mathbf{R})$ we will write in such places \hat{m} instead of m , \hat{n} instead of n etc.

The composition of morphisms is defined by the formula

$$(\widehat{k}, (\widehat{m}, f)) \circ (\widehat{m}, (\widehat{n}, g)) = (\widehat{k}, (\widehat{n}, f \widehat{\circ} g))$$

and the identity morphisms by the formula

$$[\mathbf{2015.08.22.eq4}] Id_{\widehat{n}} = (\widehat{n}, (\widehat{n}, \eta_n)) = (\widehat{n}, (\widehat{n}, (x_0^n, \dots, x_{n-1}^n))) \quad (14)$$

The axioms of a category follow from Lemma 2.3(1,2).

We also consider the precategory F with the set of objects $Ob(F) = \mathbf{N}$ and the set of morphisms given by

$$Mor(F) = \coprod_{n,m} F(n, m)$$

The domain and codomain functions are obvious and the identity morphisms and compositions are defined by identity functions and compositions of functions between the standard finite sets.

The identity function on \mathbf{N} together with the mapping

$$(\Phi_{\mathbf{R}})_{Mor} : (n, (m, f)) \mapsto (\widehat{m}, (\widehat{n}, \phi_{\mathbf{R}}(f)))$$

defines, in view of the obvious equality $\phi_{\mathbf{R}}(Id_{\widehat{n}}) = \eta_n$ and Lemma 2.3(3), a functor

$$\Phi_{\mathbf{R}} : F^{op} \rightarrow C(\mathbf{R})$$

(i.e., a contravariant functor from F to $C(\mathbf{R})$).

Remark 4.1 [2015.08.19.rem1] The pair $(C(\mathbf{R})^{op}, \Phi_{\mathbf{R}}^{op})$ is an algebraic theory in the sense of Lawvere (cf. [?, Definition on p.62]). It should be possible to prove, as noted in [?, after Theorem 3.3], that the function $\mathbf{R} \mapsto (C(\mathbf{R})^{op}, \Phi_{\mathbf{R}}^{op})$ is an equivalence between pre-algebraic theories and algebraic theories.

For $n \in \mathbf{N}$, let $ft(\widehat{n+1}) = \widehat{n}$ and let $ft(\widehat{0}) = \widehat{0}$.

For $n \in \mathbf{N}$, let $p_{\widehat{n+1}} : \widehat{n+1} \rightarrow \widehat{n}$ be the morphism $(\widehat{n+1}, (\widehat{n}, \iota_n^1))$, where we have used our coercion convention to write ι_n^1 instead of $\phi_{\mathbf{R}}(\iota_n^1)$. In the sequence notation we have

$$[\mathbf{2015.08.24.eq6}] p_{\widehat{n+1}} = (\widehat{n}, (\widehat{n+1}, (x_0^{n+1}, \dots, x_{n-1}^{n+1}))) \quad (15)$$

Let $p_{\widehat{0}} = Id_{\widehat{0}}$.

Lemma 4.2 [2015.07.24.11] *One has:*

1. Let $f = (\widehat{m}, (\widehat{n+1}, ff))$, where $ff = (f_0, \dots, f_n)$, be a morphism. Then

$$f \circ p_{\widehat{n+1}} = (\widehat{m}, (\widehat{n}, (f_0, \dots, f_{n-1})))$$

2. Let $f = (\widehat{m}, (\widehat{n}, ff))$, where $ff = (f_0, \dots, f_{n-1})$, be a morphism. Then

$$p_{\widehat{m+1}} \circ f = (\iota_m^1(f_0), \dots, \iota_m^1(f_{n-1}))$$

Remark 4.3 [2015.08.18.rem2] In the second assertion of the lemma we use our coercion convention in the right hand side of the equality. The full form of the expressions $\iota_m^1(f_i)$ that we use there are $bind(\phi_{\mathbf{R}}(\iota_m^1))(f_i)$.

Proof: The first assertion follows by unfolding definitions from (4).

The second asserting follows by unfolding definitions from (3).

Lemma 4.4 [2015.08.20.11] *For any $n \in \mathbf{N}$ the square*

$$\begin{array}{ccc}
 \widehat{n+1} & \xrightarrow{(\widehat{n+1}, (\widehat{1}, (x_n^{n+1})))} & \mathbf{1} \\
 \text{[2015.08.20.eq3]} \downarrow & & \downarrow p_{\widehat{1}} \\
 \widehat{n} & \xrightarrow{(\widehat{n}, (\widehat{0}, ()))} & \mathbf{0}
 \end{array} \tag{16}$$

is a pull-back square in $C(\mathbf{R})$.

Proof: Since $\widehat{0}$ is a final object, we need to check that for any m the function

$$\text{Hom}_{C(\mathbf{R})}(\widehat{m}, \widehat{n+1}) \rightarrow \text{Hom}_{C(\mathbf{R})}(\widehat{m}, \widehat{n}) \times \text{Hom}_{C(\mathbf{R})}(\widehat{m}, \widehat{1})$$

given by $(\widehat{m}, (\widehat{n+1}, f)) \mapsto (((\widehat{m}, (\widehat{n+1}, f)) \circ p_{\widehat{n+1}}), ((\widehat{m}, (\widehat{n+1}, f)) \circ (x_n^{n+1})))$, is a bijection. This function is isomorphic to the function

$$f \mapsto ((\iota_n^1 \widehat{\circ} f), (x_n^{n+1} \widehat{\circ} f))$$

that can be further computed using (4) and (6) to be equal to the function

$$(f_0, \dots, f_n) \rightarrow ((f_0, \dots, f_{n-1}), (f_n))$$

that is clearly a bijection.

Given a morphism $f = (\widehat{m}, (\widehat{n}, ff)) : \widehat{m} \rightarrow \widehat{n}$ in $C(\mathbf{R})$ set

$$\text{[2015.08.26.eq1]} f^*(\widehat{n+1}) = \widehat{m+1} \tag{17}$$

and define a morphism $q(f) : \widehat{m+1} \rightarrow \widehat{n+1}$ as

$$\text{[2015.08.22.eq1]} q(f) = (\widehat{m+1}, (\widehat{n+1}, qq(ff))) \tag{18}$$

where $qq(ff)$ is defined in (12).

Lemma 4.5 [2015.08.18.14] *For any f as above the square*

$$\begin{array}{ccc}
 \widehat{m+1} & \xrightarrow{q(f)} & \widehat{n+1} \\
 \text{[2015.08.20.eq2]} \downarrow & & \downarrow p_{\widehat{n+1}} \\
 \widehat{m} & \xrightarrow{f} & \widehat{n}
 \end{array} \tag{19}$$

is a pull-back square.

Proof: Let us show first that this square commutes. We need to show that for each $i = 0, \dots, n-1$ one has

$$(\iota_n^1 \widehat{\circ} q q(f f))(i) = (f f \widehat{\circ} \iota_m^1)(i)$$

We have

$$(\iota_n^1 \widehat{\circ} q q(f f))(i) = q q(f f)(i) = \iota_m^1(f f(i))$$

where the first equality is by (4) and the second equality by definition of $q q(f f)$. On the other hand

$$(f f \widehat{\circ} \iota_m^1)(i) = \iota_m^1(f f(i))$$

by (2). To show that the square (19) is a pull-back square let us consider the diagram

$$\begin{array}{ccccc} \widehat{m+1} & \xrightarrow{q(f)} & \widehat{n+1} & \xrightarrow{(\widehat{n+1}, \widehat{1}, (x_n^{n+1}))} & \widehat{1} \\ p_{\widehat{m+1}} \downarrow & & p_{\widehat{n+1}} \downarrow & & \downarrow p_{\widehat{1}} \\ \widehat{m} & \xrightarrow{f} & \widehat{n} & \xrightarrow{(\widehat{n}, \widehat{0}, ())} & 0 \end{array}$$

We have

$$q(f) \circ (\widehat{n+1}, \widehat{1}, (x_n^{n+1})) = (\widehat{m+1}, \widehat{1}, (x_n^{n+1}) \widehat{\circ} q q(f f)) = (\widehat{m+1}, \widehat{1}, (x_m^{m+1})) \quad (20)$$

where the second equality is by (6). Therefore the outer square of this diagram is of the form (16) and therefore is a pull-back square by Lemma 4.4. The right hand side square of this diagram is of the same form and therefore is a pull-back square as well. The left hand side square commutes. Therefore the left hand side square is a pull-back square by the general properties of pull-back squares.

Problem 4.6 [2015.08.20.probl] *To construct a structure of a C-system on the category $C(\mathbf{R})$.*

Construction 4.7 [2015.08.20.constr1] For the definition of the C-system that we will use see [?, Def. 2.1 and Def. 2.3].

The length function is the identity function $Ob(C(\mathbf{R})) \rightarrow \mathbf{N}$.

The object pt is $\widehat{0}$.

The ft function was defined above.

The morphisms $p_{\widehat{n}}$ were defined above.

The objects $f^*(X)$ were defined above.

One defines $q(f, X)$ as $q(f)$.

The first four properties of [?, Def. 2.1] are obvious.

The fifth property is the commutativity part of Lemma 4.5.

The sixth property is obvious.

To prove the seventh property consider $g = (\widehat{k}, (\widehat{m}, gg))$ and $f = (\widehat{m}, (\widehat{n}, ff))$. We need to show that

$$q(g) \circ q(f) = q(g \circ f)$$

Consider the diagram

$$\begin{array}{ccccccc}
\widehat{k+1} & \xrightarrow{q(g)} & \widehat{m+1} & \xrightarrow{q(f)} & \widehat{n+1} & \xrightarrow{(\widehat{n+1}, (\widehat{1}, (x_n^{n+1})))} & \widehat{1} \\
p_{\widehat{k+1}} \downarrow & & p_{\widehat{m+1}} \downarrow & & p_{\widehat{n+1}} \downarrow & & \downarrow p_{\widehat{1}} \\
\widehat{k} & \xrightarrow{g} & \widehat{m} & \xrightarrow{f} & \widehat{n} & \xrightarrow{(\widehat{n}, (\widehat{0}, ()))} & \widehat{0}
\end{array}$$

Since the rightmost square is a pull-back square it is sufficient to prove that

$$q(g) \circ q(f) \circ p_{\widehat{n+1}} = q(g \circ f) \circ p_{\widehat{n+1}}$$

and

$$q(g) \circ q(f) \circ (\widehat{n+1}, (\widehat{1}, (x_n^{n+1}))) = q(g \circ f) \circ (\widehat{n+1}, (\widehat{1}, (x_n^{n+1})))$$

The first equality follows from the commutativity of the left hand side and central squares of the diagram and the commutativity of the square (19) for $g \circ f$. For the second equality we have

$$q(g) \circ q(f) \circ (\widehat{n+1}, (\widehat{1}, (x_n^{n+1}))) = q(g) \circ (\widehat{m+1}, (\widehat{1}, (x_m^{m+1}))) = (\widehat{k+1}, (\widehat{1}, (x_k^{k+1})))$$

and

$$q(g \circ f) \circ (\widehat{n+1}, (\widehat{1}, (x_n^{n+1}))) = (\widehat{k+1}, (\widehat{1}, (x_k^{k+1})))$$

where all three equalities are of the form (20). This completes the construction of the structure of a C0-system on $C(\mathbf{R})$. To extend it to the structure of a C-system we can apply [?, Prop. 2.4] and Lemma 4.5.

Recall that for a C-system CC , and object Γ of CC such that $l(\Gamma) \geq i$ we let $p_{\Gamma, i}$ denote the morphism $\Gamma \rightarrow ft^i(\Gamma)$ defined inductively as

$$p_{\Gamma, 0} = Id_{\Gamma}$$

$$p_{\Gamma, i+1} = p_{\Gamma} \circ p_{ft(\Gamma), i}$$

For Γ as above and $f : \Gamma' \rightarrow ft^i(\Gamma)$ we let $f^*(\Gamma, i)$ and

$$q(f, \Gamma, i) : f^*(\Gamma, i) \rightarrow \Gamma$$

define a pair of an object and a morphism defined inductively as

$$f^*(\Gamma, 0) = \Gamma' \quad q(f, \Gamma, 0) = f$$

$$f^*(\Gamma, i+1) = q(f, ft(\Gamma), i)^*(\Gamma) \quad q(f, \Gamma, i+1) = q(q(f, ft(\Gamma), i), \Gamma)$$

For Γ, Γ' in a C-system let us write $\Gamma \leq \Gamma'$ if $l(\Gamma) \leq l(\Gamma')$ and $\Gamma = ft^{l(\Gamma')-l(\Gamma)}(\Gamma')$. We will write $\Gamma < \Gamma'$ if $\Gamma \leq \Gamma'$ and $l(\Gamma) < l(\Gamma')$.

If Γ' is over Γ we will denote by $p_{\Gamma', \Gamma}$ the morphism

$$p_{\Gamma', l(\Gamma')-l(\Gamma)} : \Gamma' \rightarrow \Gamma$$

If Γ' and Γ'' are over Γ then we have morphisms

$$p_{\Gamma', \Gamma} : \Gamma' \rightarrow \Gamma$$

$$p_{\Gamma'',\Gamma} : \Gamma'' \rightarrow \Gamma$$

and we say that a morphism $f : \Gamma' \rightarrow \Gamma''$ is over Γ if

$$f \circ p_{\Gamma,\Gamma''} = p_{\Gamma,\Gamma'}$$

If Γ' is an object over Γ and $f : \Delta \rightarrow \Gamma$ is a morphism then let us denote simply by $f^*(\Gamma')$ the object $f^*(\Gamma', n)$ where $n = l(\Gamma') - l(\Gamma)$. Note that n can always be inferred from f and Γ' .

Similarly we will write simply $q(f, \Gamma)$ for $q(f, \Gamma, n)$ since n can be inferred as $l(\Gamma) - l(\text{codom}(f))$.

Lemma 4.8 [2015.08.23.11a] *Let Γ', Γ'' be objects over Γ , $a : \Gamma' \rightarrow \Gamma''$ a morphism over Γ and $f : \Delta \rightarrow \Gamma$ a morphism. Then there is a unique morphism $f^*(a) : f^*(\Gamma') \rightarrow f^*(\Gamma'')$ over Δ such that the square*

$$\begin{array}{ccc} f^*(\Gamma') & \xrightarrow{q(f,\Gamma')} & \Gamma' \\ f^*(a) \downarrow & & \downarrow a \\ f^*(\Gamma'') & \xrightarrow{q(f,\Gamma'')} & \Gamma'' \end{array}$$

commutes.

Proof: We have a square

$$\begin{array}{ccc} f^*(\Gamma'') & \xrightarrow{q(f,\Gamma'')} & \Gamma'' \\ \text{[2015.08.23.eq3]} \downarrow p_{f^*(\Gamma''),\Delta} & & \downarrow p_{\Gamma'',\Delta} \\ \Delta & \xrightarrow{f} & \Gamma \end{array} \quad (21)$$

This square is a pull-back square as a vertical composition of $l(\Gamma'') - l(\Gamma)$ pull-back squares. We define $f^*(a)$ as the unique morphism such that

$$\text{[2015.08.23.eq1]} f^*(a) \circ q(f, \Gamma'') = q(f, \Gamma') \circ a \quad (22)$$

and

$$\text{[2015.08.23.eq2]} f^*(a) \circ p_{f^*(\Gamma''),\Delta} = p_{f^*(\Gamma'),\Delta} \quad (23)$$

The first of these two equalities is equivalent to the commutativity of the square (21) and the second to the condition that $f^*(a)$ is a morphism over Δ .

Lemma 4.9 [2015.08.29.12] *Let $a : \Gamma' \rightarrow \Gamma''$ be a morphism over Γ , Γ''' another object over Γ and suppose that a is a morphism over Γ''' . Let $f : \Delta \rightarrow \Gamma$ be a morphism. Then one has*

$$\text{[2015.08.29.eq2]} f^*(a) = q(f, \Gamma''')^*(a) \quad (24)$$

Proof: The morphisms involved in the proof can be seen on the diagram

$$\begin{array}{ccc}
f^*(\Gamma') & \xrightarrow{q(f, \Gamma')} & \Gamma' \\
f^*(a) \downarrow & & \downarrow a \\
f^*(\Gamma'') & \xrightarrow{q(f, \Gamma'')} & \Gamma'' \\
p_{f^*(\Gamma''), f^*(\Gamma''')} \downarrow & & \downarrow p_{\Gamma'', \Gamma'''} \\
f^*(\Gamma''') & \xrightarrow{q(f, \Gamma''')} & \Gamma''' \\
p_{f^*(\Gamma'''), \Delta} \downarrow & & \downarrow p_{\Gamma''', \Gamma} \\
\Delta & \xrightarrow{f} & \Gamma
\end{array}$$

The right hand side of (24) is a morphism over $f^*(\Gamma''')$ and therefore a morphism over Δ . It remains to verify that it satisfies equation (22). This follows immediately from its definition.

Let us describe the constructions introduced above in the case of $C(\mathbf{R})$. Note that our wide-hat notation that distinguishes the places in formulas where natural numbers are used as objects of $C(\mathbf{R})$ allows us to avoid the ambiguity that might have arisen otherwise. For example $p_{n', n}$ could be understood either as the canonical morphism $n' \rightarrow n$ using the notation $p_{\Gamma', \Gamma}$ introduced above or as the canonical morphism $n' \rightarrow n' - n$ using the notation $p_{\Gamma, i}$ that we have used in [?]. The use of the wide-hat diacritic allows to distinguish between $p_{\widehat{n}, \widehat{n}}$ - a morphism $\widehat{n}' \rightarrow \widehat{n}$, and $p_{\widehat{n}', n}$ - a morphism $\widehat{n}' \rightarrow \widehat{n' - n}$.

Lemma 4.10 [2015.08.22.16] *Let $n, i \in \mathbf{N}$.*

1. *One has*

$$(a) \ p_{\widehat{n+i}, i} = (\widehat{n+i}, (\widehat{n}, (x_0^n \dots, x_{n-i-1}^n))) = (\widehat{n+i}, (\widehat{n}, \iota_n^{n+i})),$$

$$(b) \text{ for } g = (\widehat{n}', (\widehat{n+i}, (g_0, \dots, g_{n+i-1}))) \text{ one has } g \circ p_{\widehat{n+i}, i} = (\widehat{n}', (\widehat{n}, (g_0, \dots, g_{n-1}))),$$

2. *for $f = (\widehat{m}, (\widehat{n}, ff))$ one has*

$$f^*(\widehat{n+i}, i) = m+i$$

and

$$q(f, \widehat{n+i}, i) = q^i(f) = (\widehat{m+i}, (\widehat{n+i}, qq^i(ff)))$$

where $q^i(f)$ is the i -th iteration of q applied to f .

Proof: All three assertions are proved by induction on i . For the first assertion both parts are proved by induction simultaneously. One has

1. the case $i = 0$ is (14) and the identity axiom of the category $C(\mathbf{R})$,

2. for the successor of i we have

$$p_{\widehat{n+i+1}, i+1} = p_{\widehat{n+i+1}} \circ p_{\widehat{n+i}, i} = (\widehat{n+i+1}, (\widehat{n}, (x_0^n, \dots, x_{n-1}^n)))$$

where the second equality is by the second part of the inductive assumption. For the inductive step in the second part we have

$$\begin{aligned} (\widehat{n}', (\widehat{n+i+1}, (g_0, \dots, g_{n+i}))) \circ p_{\widehat{n+i+1}, i+1} &= (\widehat{n}', (\widehat{n+i+1}, (g_0, \dots, g_{n+i}))) \circ p_{\widehat{n+i+1}} \circ p_{\widehat{n+i}, i} = \\ &(\widehat{n}', (\widehat{n+i}, (g_0, \dots, g_{n+i-1}))) \circ p_{\widehat{n+i}, i} = (\widehat{n}', (\widehat{n}, (g_0, \dots, g_{n-1}))) \end{aligned}$$

The proof of the first part of the second assertion is obvious. For the second part we have:

1. for $i = 0$ the assertion is obvious,
2. for the successor of i we have

$$q(f, \widehat{n}, i+1) = q(q(f, \widehat{n-1}, i)) = q(q^i(f))$$

and

$$q(q^i(f)) = q((\widehat{m+i}, (\widehat{n+i}, qq^i(ff)))) = (\widehat{m+i+1}, (\widehat{n+i+1}, qq^{i+1}(ff)))$$

Lemma 4.11 [2015.08.22.17] *Let $f = (\widehat{n}, (\widehat{n+1}, (f_0, \dots, f_n)))$. Then $f \circ p_{\widehat{n+1}} = Id_{\widehat{n}}$ if and only if $f_i = x_i^{n+1}$ for $i = 0, \dots, n-1$.*

Proof: It follows immediately from Lemma 2.8.

Lemma 4.12 [2015.09.09.11] *Let $f = (\widehat{m}, (\widehat{n}, (f_0, \dots, f_{n-1})))$ where $n > 0$. Then one has*

$$s_f = (\widehat{m}, (\widehat{m+1}, (x_0^m, \dots, x_{m-1}^m, f_{n-1})))$$

Proof: By [?, Definition 2.3(2)] we have that

$$s_f \circ p_{\widehat{m+1}} = Id_{\widehat{m}}$$

Therefore, by Lemma 4.11, it is of the form $(\widehat{m}, (\widehat{m+1}, (x_0^m, \dots, x_{m-1}^m, sf)))$ for some $sf \in R(m)$. By [?, Definition 2.3(3)] we have $f = s_f \circ q(ft(f))$ where $ft(f) = f \circ p_{\widehat{n}}$. By Lemma 4.2(1) we have $ft(f) = (\widehat{m}, (\widehat{n-1}, (f_0, \dots, f_{n-1})))$ and by (18) and (12) we have that

$$q(ft(f)) = (\widehat{m+1}, (\widehat{n}, (\iota_m^1(f_0), \dots, \iota_m^1(f_{n-2}), x_m^{m+1})))$$

Therefore, by (3) we should have

$$(f_0, \dots, f_{n-1}) = (\iota_m^1(f_0), \dots, \iota_m^1(f_{n-2}), x_m^{m+1}) \widehat{\circ} (x_0^m, \dots, x_{m-1}^m, sf)$$

which is equivalent to

$$f_i = (x_0^m, \dots, x_{m-1}^m, sf)(\iota_m^1(f_i))$$

for $i = 0, \dots, n-2$ and

$$f_{n-1} = (x_0^m, \dots, x_{m-1}^m, sf)(x_m^{m+1})$$

The first series of equalities follow for any sf from (2) since

$$\iota_m^1 \widehat{\circ} (x_0^m, \dots, x_{m-1}^m, sf) = (x_0^m, \dots, x_{m-1}^m)$$

The last equality gives us, by (6) that $f_{n-1} = sf$.

Recall from [?] that for a C-system CC one defines $\widetilde{Ob}(CC)$ as the subset of $Mor(CC)$ which consists of morphisms s of the form $ft(X) \rightarrow X$ such that $l(X) > 0$ and $s \circ p_X = Id_{ft(X)}$.

Lemma 4.13 [2015.08.24.11] *Let $f = (\widehat{m}, (\widehat{n}, ff))$ and let $s = (\widehat{n}, (\widehat{n+1}, ss))$ be an element of \widetilde{Ob} . Then one has*

$$f^*(s) = (\widehat{m}, (\widehat{m+1}, fs))$$

where

$$fs = (x_0^m, \dots, x_{m-1}^m, ff(ss_n))$$

Proof: The fact that the first m terms of the sequence representation of fs have the required form follows from Lemma 4.11. It remains to prove that

$$fs_m = ff(ss_n)$$

By (18) we have

$$q(f, \widehat{n+1}) = q(f) = (\widehat{m+1}, (\widehat{n+1}, qq(ff)))$$

The morphism $f^*(s)$, as a morphism over \widehat{m} is defined by the equation

$$f^*(s) \circ q(f) = f \circ s$$

which is equivalent to

$$qq(ff) \widehat{\circ} fs = (\iota_m^1(f_0), \dots, \iota_m^1(f_{n-1}), x_m^{m+1}) \widehat{\circ} fs = ss \widehat{\circ} ff$$

Considering only the last terms of the corresponding sequences we get

$$fs_m = fs(x_m^{m+1}) = ff(ss_n)$$

where the first equality is (5).

Lemma 4.14 [2015.08.29.11] *Let $f = (\widehat{m}, (\widehat{n}, ff))$ and let $s = (\widehat{n+i}, (\widehat{n+i+1}, ss))$ be an element of \widetilde{Ob} . Then one has*

$$[2015.08.29.eq1] f^*(s) = (\widehat{m+i}, (\widehat{m+i+1}, (x_0^{m+i}, \dots, x_{m+i-1}^{m+i}, (qq^i(ff))(ss_{n+i})))) \quad (25)$$

Proof: The morphisms involved in the proof can be seen on the following diagram

$$\begin{array}{ccc} \widehat{m+i} & \xrightarrow{q^i(f)} & \widehat{n+i} \\ f^*(s) \downarrow & & \downarrow s \\ \widehat{m+i+1} & \xrightarrow{q^{i+1}(f)} & \widehat{n+i+1} \\ p_{m+i+1, i+1} \downarrow & & \downarrow p_{n+i+1, i+1} \\ \widehat{m} & \xrightarrow{f} & \widehat{n} \end{array}$$

The morphism s is a morphism to $Id_{\widehat{n+i}}$ over $\widehat{n+i}$. Therefore, we may apply Lemma 4.9 obtaining the equality

$$f^*(s) = (q^i(f))^*(s)$$

On the other hand by Lemma 4.10(2) we have $q^i(f) = (\widehat{m+i}, (\widehat{n+i}, qq^i(ff)))$ and by Lemma 4.13 we have

$$(\widehat{m+i}, (\widehat{n+i}, qq^i(ff)))^* s = (\widehat{m+i}, (\widehat{m+i+1}, (x_0^{m+i}, \dots, x_{m+i-1}^{m+i}, (qq^i(ff))(ss_n)))).$$

Another operation that we would like to have an explicit form of is operation δ . For C-system CC and an object Γ in CC one defines $\delta_\Gamma : \Gamma \rightarrow p_\Gamma^*(\Gamma)$ as the unique morphism over Γ such that

$$[\mathbf{2015.08.24.eq10}] \delta_\Gamma \circ q(p_\Gamma, \Gamma) = Id_\Gamma \quad (26)$$

Lemma 4.15 [$\mathbf{2015.08.24.15}$] *Let $n > 0$ then*

$$\delta_{\widehat{n}} = (n, (n+1, (x_0^n, \dots, x_{n-1}^n, x_{n-1}^n)))$$

Proof: In view of Lemma 4.11, we have $\delta_n = (n, (n+1, (x_0^n, \dots, x_{n-1}^n, dn)))$ for some $dn \in R(n)$. By Lemma 2.10 we have

$$q(p_{\widehat{n}}) = (\widehat{n+1}, (\widehat{n}, \partial_n^{n-1})) = (\widehat{n+1}, (\widehat{n}, (x_0^{n+1}, \dots, x_{n-2}^{n+1}, x_n^{n+1})))$$

and the defining equation (26) gives us

$$(n, (n+1, (x_0^n, \dots, x_{n-1}^n, dn))) \circ (\widehat{n+1}, (\widehat{n}, (x_0^{n+1}, \dots, x_{n-2}^{n+1}, x_n^{n+1}))) = (n, (n, (x_0^n, \dots, x_{n-1}^n)))$$

i.e.,

$$(x_0^{n+1}, \dots, x_{n-2}^{n+1}, x_n^{n+1}) \widehat{\circ} (x_0^n, \dots, x_{n-1}^n, dn) = (x_0^n, \dots, x_{n-1}^n)$$

and comparing the last terms of the sequences we get

$$dn = (x_0^n, \dots, x_{n-1}^n, dn)(x_n^{n+1}) = x_{n-1}^n$$

Problem 4.16 *To construct a bijection*

$$[\mathbf{2015.08.24.eq9}] mb_{\mathbf{R}} : \widetilde{Ob}(C(\mathbf{R})) \rightarrow \coprod_{n \in \mathbf{N}} R(n) \quad (27)$$

Construction 4.17 [$\mathbf{2015.08.22.constr3}$] For $s = (n, (n+1, ss))$ define

$$mb_{\mathbf{R}}(s) = (n, ss_n)$$

To show that this is a bijection let us construct the inverse bijection. For $o \in R(n)$ set

$$mb_{\mathbf{R}}^{\dagger}(n, o) = (n, (n+1, (x_0^n, \dots, x_{n-1}^n, o)))$$

The fact that these pas a mutually inverse follows easily from Lemma 4.11.

Our next goal is to describe operations T' , \widetilde{T}' , S' , \widetilde{S}' and δ' obtained from operations T , \widetilde{T} , S , \widetilde{S} and δ that were introduced at the end of Section 3 in [?] through transport by means of the bijection (27).

Let us first recall the definition of operations T , \widetilde{T} , S , \widetilde{S} and δ associated with a general C-system CC .

Definition 4.18 [$\mathbf{2015.08.26.def1}$] *Let CC be a C-system. We will write Ob for $Ob(CC)$ and \widetilde{Ob} for $\widetilde{Ob}(CC)$.*

1. Operation T is defined on the set

$$T_{dom} = \{\Gamma, \Gamma' \in Ob \mid l(\Gamma) > 0 \text{ and } \Gamma' > ft(\Gamma)\}$$

and takes values in Ob . For $(\Gamma, \Gamma') \in T_{dom}$ one has

$$T(\Gamma, \Gamma') = p_{\Gamma}^*(\Gamma')$$

2. Operation \tilde{T} is defined on the set

$$\tilde{T}_{dom} = \{\Gamma \in Ob, s \in \tilde{Ob} \mid l(\Gamma) > 0 \text{ and } \partial(s) > ft(\Gamma)\}$$

and takes values in \tilde{Ob} . For $(\Gamma, s) \in \tilde{T}_{dom}$ one has

$$\tilde{T}(\Gamma, s) = p_{\Gamma}^*(s)$$

3. Operation S is defined on the set

$$S_{dom} = \{r \in \tilde{Ob}, \Gamma \in Ob \mid \Gamma > \partial(r)\}$$

and takes values in Ob . For $(r, \Gamma) \in S_{dom}$ one has

$$S(r, \Gamma) = r^*(\Gamma)$$

4. Operation \tilde{S} is defined on the set

$$\tilde{S}_{dom} = \{r, s \in \tilde{Ob} \mid \partial(s) > \partial(r)\}$$

and takes values in \tilde{Ob} . For $(r, s) \in \tilde{S}_{dom}$ one has

$$S(r, s) = r^*(s)$$

5. Operation δ is defined on the set

$$\delta_{dom} = \{\Gamma \in Ob \mid l(\Gamma) > 0\}$$

and takes values in \tilde{Ob} . For $\Gamma \in \delta_{dom}$, $\delta(\Gamma)$ is the unique morphism $\Gamma \rightarrow p_{\Gamma}^*(\Gamma)$ over Γ such that

$$\delta_{\Gamma} \circ q(p_{\Gamma}, \Gamma) = Id_{\Gamma}$$

Theorem 4.19 [2015.08.26.th1] Let $Ob = Ob(C(\mathbf{R}))$ and let \tilde{Ob}' be the right hand side of (27). One has:

1. Operation T' is defined on the set

$$T'_{dom} = \{\hat{m}, \hat{n} \in Ob \mid m > 0 \text{ and } n > m - 1\}$$

and is given by

$$T'(\hat{m}, \hat{n}) = \widehat{n + 1}$$

2. Operation \widetilde{T}' is defined on the set

$$\widetilde{T}'_{dom} = \{\widehat{m} \in Ob, (n, s) \in \widetilde{Ob}' \mid m > 0 \text{ and } n + 1 > m - 1\}$$

and is given by

$$\widetilde{T}'(\widehat{m}, (n, s)) = (n + 1, \partial_n^{m-1}(s))$$

3. Operation S' is defined on the set

$$S'_{dom} = \{(m, r) \in \widetilde{Ob}', \widehat{n} \in Ob \mid n > m + 1\}$$

and is given by

$$S'((m, r), \widehat{n}) = \widehat{n - 1}$$

4. Operation \widetilde{S}' is defined on the set

$$\widetilde{S}'_{dom} = \{(m, r) \in \widetilde{Ob}', (n, s) \in \widetilde{Ob}' \mid n > m\}$$

and is given by

$$\widetilde{S}'((m, r), (n, s)) = (n - 1, \sigma_{\mathbf{R}}(r, s))$$

5. Operation δ' is defined on the subset

$$\delta'_{dom} = \{\widehat{n} \in Ob \mid n > 0\}$$

and is given by

$$\delta'(\widehat{n}) = (n, x_{n-1}^n)$$

Proof: We have:

1. Operation T' is the same as operation T for $C(\mathbf{R})$ since \widetilde{Ob} is not involved in it. The form of T'_{dom} is obtained by unfolding definitions and the formula for the operation itself follows from (17).
2. Operation \widetilde{T}' is defined on the set of pairs $(\widehat{m} \in Ob, (n, s) \in \widetilde{Ob}')$ such that $m > 0$ and $\partial(mb_{\mathbf{R}}^1(n, s)) > m - 1$. Since $\partial(mb_{\mathbf{R}}^1(n, s)) = n + 1$ we obtain the required domain of definition. The formula by the operation itself is obtained immediately by combining Lemma 4.14 and Lemma 2.10.
3. Operation S' is defined on the set of pairs $((m, r) \in \widetilde{Ob}', \widehat{n} \in Ob)$ such that $n > \partial(mb_{\mathbf{R}}^1(m, r))$. Since $\partial(mb_{\mathbf{R}}^1(m, r)) = m + 1$ we obtained the required domain of definition. The operation itself is given by

$$\begin{aligned} S'((m, r), n) &= (mb_{\mathbf{R}}^1(m, r))^*(\widehat{n}) = ((\widehat{m}, (\widehat{m + 1}, (x_0^m, \dots, x_{m-1}^m, r))))^*(\widehat{n}) = \\ &= n + \widehat{m} - (\widehat{m + 1}) = \widehat{n - 1} \end{aligned}$$

4. Operation \widetilde{S}' is defined on the set of pairs $(m, r), (n, s) \in \widetilde{Ob}'$ such that $\partial(mb_{\mathbf{R}}^1(n, s)) > \partial(mb_{\mathbf{R}}^1(m, r))$ which is equivalent to $n > m$. The formula by the operation itself is obtained immediately by combining Lemma 4.14 with $i = n - m - 1$, Lemma 2.11 and the definition of $\sigma_{\mathbf{R}}$ in (13).

5. Operation δ' is defined on the subset $\widehat{n} \in Ob$ such that $n > 0$ and is given by

$$\delta'(\widehat{n}) = mb_{\mathbf{R}}(\delta(\widehat{n})) = mb_{\mathbf{R}}(\widehat{n}, (\widehat{n+1}, (x_0^n, \dots, x_{n-1}^n, x_{n-1}^n))) = (n, x_{n-1}^n)$$

Remark 4.20 [2015.08.29.rem1] The description of operations T' , \widetilde{T}' , S , \widetilde{S}' and δ given in Theorem 4.19 uses only three operations on the sets $R(n)$:

1. for all $m, n \in \mathbf{N}$, $n > m$, a function $\sigma_{m,n} : R(m) \times R(n) \rightarrow R(n-1)$ given by

$$\sigma_{m,n}(r, s) = @\sigma(m, r, n, s) = (x_0^{n-1}, \dots, x_{m-1}^{n-1}, \iota_m^{n-m-1}(r), x_m^{n-1}, \dots, x_{n-2}^{n-1})(s)$$

2. for all $i, n \in \mathbf{N}$, $n \geq i$, a function $\partial_n^i : R(n) \rightarrow R(n+1)$

3. for all $n \in \mathbf{N}$ an element $x_n \in R(n+1)$ given as $x_n = \eta_{n+1}(n)$.

Therefore, the conjectural equivalence between C-systems and B-systems implies that it is possible to construct $C(\mathbf{R})$ starting with a family of sets $R(n)$ equipped with operations:

$$\sigma_{m,n} : R(m) \times R(n) \rightarrow R(n-1)$$

$$\partial_n^i : R(n) \rightarrow R(n+1)$$

$$x_n \in R(n+1)$$

that satisfy some axioms. To avoid working with families of sets or types which might be inconvenient in simple-typed languages such as HOL we can further re-write this definition as follows. Let lR be the disjoint union of $R(n)$ for all n . Then the operations that we need to consider are:

1. a function $l : lR \rightarrow \mathbf{N}$,
2. a function $\eta : \mathbf{N} \rightarrow lR$,
3. a function $\partial : \{r \in lR, i \in \mathbf{N} \mid l(r) \geq i\} \rightarrow lR$,
4. a function $\sigma : \{r, s \in lR, l(r) > l(s)\} \rightarrow lR$,

such that

1. for all $n \in \mathbf{N}$, $l(\eta(n)) = n + 1$,
2. for all $r \in lR$, $i \in \mathbf{N}$ such that $l(r) \geq i$, $l(\partial(r, i)) = l(r) + 1$,
3. for all $r, s \in lR$ such that $l(s) > l(r)$ one has $l(\sigma(r, s)) = l(s) - 1$.

Clearly more axioms need to be added to obtain a full subcategory in the category whose objects are sets lR with operations of the form $l, \eta, \partial, \sigma$ that is equivalent to the category of Lawvere theories or equivalently pre-algebraic theories or equivalently Fiore-Plotkin-Turi substitution algebras. When all the axioms are added one would obtain a definition of an l-algebraic theory that is convenient for formalization both in set theory and in HOL.

Remark 4.21 [2015.08.29.rem1b] It would appear that there should be examples of (η, ∂, σ) -structures that do not arise from pre-algebraic theories. Indeed, it seems unclear why it should be possible to realize the action of the symmetric group on $R(n)$ using these operations since in the case of pre-algebraic theories they all seem to respect the linear ordering on the sets $stn(n)$ in some sense.

In the substitution notation of Remark 5.1, given r in $R(m)$ and E in $R(n)$,

$$\sigma(r, E) = E[r/x_m, x_m/x_{m+1}, \dots, x_{n-2}/x_{n-1}],$$

i.e., operation σ corresponds to the substitution of an expression in variables x_0, \dots, x_{m-1} for the variable x_m in an expression in variables x_0, \dots, x_n followed by a downshift of the indexes of the variables with the higher index.

The operation ∂_n^i and the constants x_n are similarly defined in terms of linear orderings.

To see how it is, nevertheless, possible to realize, for example, the permutation of x_0 and x_1 consider the following. First let, for all $i, n \in \mathbf{N}$,

$$\iota_n^i = \partial_{n+i-1}^{n+i-1} \circ \dots \circ \partial_n^n : R(n) \rightarrow R(n+i)$$

Then define for all $i, n \in \mathbf{N}$, $n \geq i+1$ an element $x_i^n \in R(n)$ by the formula

$$x_i^n = \iota_{i+1}^{n-i-1}(x_i)$$

such that, in particular, $x_n^{n+1} = x_n$.

Define now a function $\psi : R(2) \rightarrow R(2)$ by the formula

$$\psi = \partial_2^0 \circ \partial_3^0 \circ \sigma_{3,4}(x_0^3, -) \circ \sigma_{2,3}(x_1^2, -)$$

One can verify that for any pre-algebraic theory R one has $\psi = \phi_{\mathbf{R}}(\sigma)$ where σ is the permutation of 0 and 1 in $stn(2)$.

In the substitution notation this can be seen as follows:

$$\begin{aligned} \psi(E(x_0^2, x_1^2)) &= \sigma_{2,3}(x_1^2, \sigma_{3,4}(x_0^3, \partial_3^0(\partial_2^0(E(x_0^2, x_1^2)))))) = \sigma_{2,3}(x_1^2, \sigma_{3,4}(x_0^3, \partial_3^0(E(x_1^3, x_2^3)))) = \\ &= \sigma_{2,3}(x_1^2, \sigma_{3,4}(x_0^3, E(x_2^4, x_3^4))) = \sigma_{2,3}(x_1^2, E(x_2^3, x_0^3)) = E(x_1^2, x_0^2) \end{aligned}$$

Remark 4.22 [2015.08.29.rem2] The fact that the full structure of a pre-algebraic theory can be reconstructed from operations σ, ∂ and η of the form described above can also be approached via the comparison of such $\sigma\partial\eta$ -structures with the substitution algebras of [?, Definition 3.1]. Such a comparison is more straightforward since their operation ζ of the form $R(n+1) \times R(n) \rightarrow R(n)$ is the same as operation $(s, r) \mapsto \sigma_{n,n+1}(r, s)$.

5 The C-system $CC(\mathbf{R}, \mathbf{LM})$.

Let \mathbf{R} be a pre-algebraic theory and let $\mathbf{LM} = (LM, LM_{Mor})$ be a pre-sheaf of sets on the category $C(\mathbf{R})$, i.e., a functor $C(\mathbf{R})^{op} \rightarrow Sets$.

The morphism component LM_{Mor} of \mathbf{LM} is a function that sends a morphism $f = (\hat{m}, (\hat{n}, ff))$ from \hat{m} to \hat{n} in $C(\mathbf{R})$ to a function $LM_{Mor}(f) \in Fun(LM(\hat{n}), LM(\hat{m}))$. In particular we have for, each $m, n \in \mathbf{N}$ a function

$$R(n, m) \rightarrow Fun(LM(\hat{n}), LM(\hat{m}))$$

given by $ff \mapsto LM_{Mor}((\widehat{m}, (\widehat{n}, ff)))$. We will use this function as a coercion so that, for $ff \in R(n, m)$ and $E \in LM(\widehat{n})$ the expression $ff(E)$ needs to be expanded into $LM_{Mor}((\widehat{m}, (\widehat{n}, ff)))(E)$ before the computation can occur (see the explanation after Lemma 2.3 for more details).

Remark 5.1 [2015.08.18.rem1] If we think of $E \in LM(\widehat{n})$ as of an expression in variables $0, \dots, n-1$ then the action of $R(n, m)$ on $LM(\widehat{n})$ can be thought of as the substitution. This analogy can be used to introduce the notation when for $ff = (f_0, \dots, f_{n-1}) \in R(n, m)$ and $E \in LM(\widehat{n})$ one writes $ff(E)$ as

$$ff(E) = E[f_0/0, \dots, f_{n-1}/n-1]$$

For example, in this notation we have

$$\partial_n^i(E) = E[0/0, \dots, i-1/i-1, i+1/i, \dots, n/n-1]$$

Similarly, for $E \in LM(\widehat{n+2})$ one has

$$\sigma_n^i(E) = E[0/0, \dots, i/i, i/i+1, \dots, n/n+1]$$

and $\iota_n^i(E)$ is “the same expression” but considered as an expression of $n+i$ variables.

Example 5.2 [2015.09.07.rem3] An important example of **LM** is given by the functor Φ given on objects by $\Phi_{Ob}(\widehat{n}) = R(n)$ and on morphisms by

$$\Phi_{Mor}((\widehat{m}, (\widehat{n}, r)))(s) = r(s)$$

We will denote this functor by the same symbol **R** as the underlying pre-algebraic theory.

This functor is isomorphic to the (contravariant) functor represented by the object $\widehat{1}$ but it is not equal to this functor since the set of elements of the form $(\widehat{n}, (\widehat{1}, r'))$ where $r' \in R(1, n)$ is isomorphic but not equal to the set $R(n)$.

Example 5.3 [2015.09.07.rem4] Another important example of a presheaf **LM** is the constant presheaf corresponding to a set S . The C-system $CC(\mathbf{R}, \mathbf{LM})$ for $\mathbf{LM} = S$ corresponds to the multi-sorted algebraic theory with a fixed set of sorts S .

Taking **LM** to be a sub-presheaf of $\mathbf{LM}_0 \times S$ one can add an extra “sorting dimension” to a theory. Cf. Remark ??.

Recall that we defined, for any pre-algebraic theory **R**, an operation σ that we can write in the form

$$\sigma_{\mathbf{R}} : \{m \in \mathbf{N}, r \in R(m), n \in \mathbf{N}, s \in R(n) \mid n > m\} \rightarrow R(n-1)$$

which is a *dependent* function, i.e., a function whose codomain depends on the argument.

For LM as above define a dependent function $\sigma_{\mathbf{LM}}$ with the domain of definitions

$$dom(\sigma_{\mathbf{LM}}) = \{m \in \mathbf{N}, r \in R(m), n \in nat, E \in LM(\widehat{n}) \mid n > m\},$$

and the codomain function

$$codom(\sigma_{\mathbf{LM}})(m, r, n, E) = LM(\widehat{n-1})$$

by the formula

$$\begin{aligned} \text{[2015.09.07.eq2]} \sigma_{\mathbf{LM}}(m, r, n, E) &= (qq^{n-m-1}(x_0^m, \dots, x_{m-1}^m, r))(E) = \\ &= (x_0^{n-1}, \dots, x_{m-1}^{n-1}, \iota_m^{n-m-1}(r), x_m^{n-1}, \dots, x_{n-2}^{n-1})(E) \end{aligned} \quad (28)$$

where the second equality is the equality of Lemma 2.11. Alternatively, one can think of $\sigma_{\mathbf{LM}}$ as of a family of functions $\sigma_{\mathbf{LM},n}$ with different codomains but we will use the dependent function viewpoint.

Of the four arguments m, r, n and E of $\sigma_{\mathbf{LM}}$ we will use m and n as implicit arguments, that is, arguments that are not written explicitly because they often can be inferred from the remaining arguments by a simple algorithmic procedure. In cases when m and n can not be inferred from r and E , which we will not encounter in this paper, the syntax of the proof assistant Coq is to write $@\sigma_{\mathbf{LM}}$ instead of $\sigma_{\mathbf{LM}}$ and to supply all four arguments explicitly.

Define a category $CC(\mathbf{R}, \mathbf{LM})$ as follows.

$$Ob(CC(\mathbf{R}, \mathbf{LM})) = \coprod_{n \in \mathbf{N}} Ob_n(\mathbf{R}, \mathbf{LM})$$

where

$$Ob_n(\mathbf{R}, \mathbf{LM}) = LM(\widehat{0}) \times \dots \times LM(\widehat{n-1})$$

Remark 5.4 [2015.08.14.rem1] In a univalent formalization based on UniMath one can define $Ob_n(\mathbf{R}, \mathbf{LM})$ as the type *forall*($i : stn\ n$), $LM(i)$. Based on this formalization elements of $Ob_n(\mathbf{R}, \mathbf{LM})$ are dependent functions with the domain $stn(n)$ and the codomain function $i \mapsto LM(\widehat{i})$. Therefore, for $X \in Ob_n$ and $i \in stn(n)$ it makes sense to write $X(i)$ for the i -th component of X . However, for typographical reasons, we will sometimes write X_i instead of $X(i)$.

Define a function l_f on $Ob(CC(\mathbf{R}, \mathbf{LM}))$ setting $l_f(n, A) = n$.

The set of morphisms of $CC(\mathbf{R}, \mathbf{LM})$ is given by

$$Mor(CC(\mathbf{R}, \mathbf{LM})) = \coprod_{\Gamma, \Gamma' \in Ob(CC(\mathbf{R}, \mathbf{LM}))} R(l_f(\Gamma'), l_f(\Gamma))$$

with the obvious domain and codomain maps.

The composition of morphisms in $CC(\mathbf{R}, \mathbf{LM})$ is defined by the formula

$$(\Gamma, (\Gamma', f)) \circ (\Gamma', (\Gamma'', g)) = (\Gamma, (\Gamma'', g \widehat{\circ} f))$$

and the identity morphisms by

$$\text{[2015.08.22.eq6]} Id_{\Gamma} = (\Gamma, (\Gamma, \eta_{l_f(\Gamma)})) = (\Gamma, (\Gamma, (x_0^n, \dots, x_{n-1}^n))) \quad (29)$$

The axioms of a category follow from Lemma 2.3(1,2).

The proof of the following lemma is omitted because of its simplicity.

Lemma 5.5 [2015.08.22.11] *The mappings*

$$\begin{aligned} \Gamma &\mapsto l_f(\Gamma) \\ (\Gamma, (\Gamma', f)) &\mapsto (l_f(\Gamma), (l_f(\Gamma'), f)) \end{aligned}$$

define a fully faithful functor

$$Fl : CC(\mathbf{R}, \mathbf{LM}) \rightarrow C(\mathbf{R})$$

Remark 5.6 [2015.09.01.rem1] For any two objects Γ, Γ' of $CC(\mathbf{R}, \mathbf{LM})$ such that $l_f(\Gamma) = l_f(\Gamma') = n$, the formula

$$can_{\Gamma, \Gamma'} = (\Gamma, (\Gamma', (x_0^n, \dots, x_{n-1}^n)))$$

defines a morphism which is clearly an isomorphism with $can_{\Gamma', \Gamma}$ being a canonical inverse. Therefore, all objects of the same length in $CC(\mathbf{R}, \mathbf{LM})$ are “canonically isomorphic”.

Remark 5.7 [2015.09.01.rem2] If $LM(\widehat{0}) = \emptyset$ then $CC(\mathbf{R}, \mathbf{LM}) = \emptyset$. On the other hand, the choice of an element y in $LM(\widehat{0})$ defines distinguished elements $\iota_0^n(y)$ in all sets $LM(\widehat{n})$ and therefore distinguished objects

$$y_n = (n, (\iota_0^0(y), \dots, \iota_0^{n-1}(y))) \in Ob(CC(\mathbf{R}, \mathbf{LM}))$$

of length n for all n . Mapping \widehat{n} to y_n defines, as one can immediately prove from the definitions, a functor $lF(y) : C(\mathbf{R}) \rightarrow CC(\mathbf{R}, \mathbf{LM})$.

This functor clearly satisfies the conditions $lF(y) \circ Fl = Id_{C(\mathbf{R})}$.

One verifies easily that the morphisms

$$can_{\Gamma, y_{l_f(\Gamma)}} : \Gamma \rightarrow lF(y)(Fl_f(\Gamma))$$

form a natural transformation. We conclude that Fl and $lF(y)$ is a pair of mutually inverse equivalences of categories.

However this equivalence is not an isomorphism unless $LM(\widehat{n}) \cong unit$ for all n .

For $\Gamma \in Ob(CC(\mathbf{R}, \mathbf{LM}))$ such that $\Gamma = (n+1, (T_0, \dots, T_n))$ denote by $ft(\Gamma)$ the object

$$ft(\Gamma) = (n, (T_0, \dots, T_{n-1}))$$

and by p_Γ the morphism

$$p_\Gamma = (\Gamma, (ft(\Gamma), \iota_n^1))$$

The following immediate analog of Lemma 4.2 holds in $CC(\mathbf{R}, \mathbf{LM})$:

Lemma 5.8 [2015.08.22.16b] *One has:*

1. Let $f = (\Gamma', (\Gamma, (f_0, \dots, f_n)))$, then

$$f \circ p_\Gamma = (\Gamma', (ft(\Gamma), (f_0, \dots, f_{n-1})))$$

2. Let $f = (ft(\Gamma'), (\Gamma, (f_0, \dots, f_n)))$, then

$$p_{\Gamma'} \circ f = (\iota_m^1(f_0), \dots, \iota_m^1(f_{n-1}))$$

Proof: Since Fl is a fully faithful functor it is sufficient to verify that the asserted equalities hold after application of Fl . This follows immediately from Lemma 4.2.

Given two objects $\Gamma' = (m, (T'_0, \dots, T'_{m-1}))$ and $\Gamma = (n + 1, (T_0, \dots, T_n))$ and a morphism $f = (\Gamma', (ft(\Gamma), ff))$ define an object $f^*(\Gamma)$ by the formula:

$$[2015.09.09.eq3old] f^*(\Gamma) = (m + 1, (T'_0, \dots, T'_{m-1}, ff(T_n))) \quad (30)$$

and a morphism $q(f, \Gamma) : f^*(\Gamma) \rightarrow \Gamma$ by the formula

$$q(f, \Gamma) = (f^*(\Gamma), (\Gamma, qq(ff)))$$

Lemma 5.9 [2015.08.26.17] *Let $\Gamma = (n, (T_0, \dots, T_{n-1}))$ where $n \geq i$, $\Gamma' = (m, (T'_0, \dots, T'_{m-1}))$ and let $(\Gamma', (ft^i(\Gamma), ff))$ be a morphism. Then one has*

$$f^*(\Gamma, i) = (m + i, (T'_0, \dots, T'_{m-1}, ff(T_{n-i}), qq(ff)(T_{n-i+1}), \dots, qq^{i-1}(ff)(T_{n-1})))$$

Proof: By straightforward induction on i .

Lemma 5.10 [2015.08.26.18] *Let $\Gamma = (n, (T_0, \dots, T_{n-1}))$ and $\Gamma' = (m, (T_0, \dots, T_{m-2}, T))$ where $n > m - 1$. Then one has*

$$p_{\Gamma'}^*(\Gamma) = (n + 1, (T_0, \dots, T_{m-2}, T, \partial_{m-1}^{m-1}(T_{m-1}), \dots, \partial_{n-1}^{m-1}(T_{n-1})))$$

Proof: One has

$$\begin{aligned} p_{\Gamma'}^*(\Gamma) &= (\Gamma', (ft(\Gamma'), \iota_{m-1}^1))^*(\Gamma) = \\ &= (n + 1, (T_0, \dots, T_{m-2}, T, \iota_{m-1}^1(T_{m-1}), \dots, qq^{n-m-2}(\iota_{m-1}^1)(T_{n-1}))) = \\ &= (n + 1, (T_0, \dots, T_{m-2}, T, \partial_{m-1}^{m-1}(T_{m-1}), \dots, \partial_{n-1}^{m-1}(T_{n-1}))) \end{aligned}$$

where the second equality is by Lemma 5.9 and the third equality is by Lemma 2.10.

Lemma 5.11 [2015.08.18.14b] *For any Γ', Γ and f as above the square*

$$\begin{array}{ccc} f^*(\Gamma) & \xrightarrow{q(f, \Gamma)} & \Gamma \\ [2015.08.22.eq2]_{(\Gamma)} \downarrow & & p_{\Gamma} \downarrow \\ \Gamma' & \xrightarrow{f} & ft(\Gamma) \end{array} \quad (31)$$

is a pull-back square.

Proof: Since Fl of Lemma 5.5 a fully faithful functor it is sufficient to show that the image of the square (31) under this functor is a pull-back square. This image is the square

$$\begin{array}{ccc} m + 1 & \xrightarrow{q(f)} & n + 1 \\ p_{m+1} \downarrow & & p_{n+1} \downarrow \\ m & \xrightarrow{f} & n \end{array}$$

which is a pull-back square by Lemma 4.5.

Problem 5.12 [2015.08.17.probl] *To construct the structure of a C-system on the category $CC(\mathbf{R}, \mathbf{LM})$.*

Construction 5.13 [2015.08.17.constr1] The length function l_f is already defined.

The object pt is the only object of length 0.

The function ft on objects of length > 0 is already defined. We set $ft(pt) = pt$.

The canonical morphisms p_Γ have already been defined for elements of length > 0 and we define p_{pt} to be the identity morphism.

The operation f^* on objects and morphisms $q(f, \Gamma)$ is already defined.

The first four conditions of [?, Definition 2.1] are obvious.

The fifth condition is the commutativity of the squares (31) which is a part of Lemma 5.11.

The sixth condition has two parts. One is that $Id_{ft(\Gamma)}^*(\Gamma) = \Gamma$ which follows from the identity morphism property of the functor \mathbf{LM} . The second part is that $q(Id_{ft(\Gamma)}, \Gamma) = Id_\Gamma$ which follows from the first part, the fact that Fl is fully faithful and the corresponding property of the C-system $C(\mathbf{R})$.

The seventh conditions also has two parts. The first one is that for $g : \Gamma'' \rightarrow \Gamma'$ one has $g^*(f^*(\Gamma)) = (g \circ f)^*(\Gamma)$. If $\Gamma'' = (T_0'', \dots, T_{k-1}'')$ and $g = (\Gamma', (\Gamma', gg))$ then we have

$$g^*(f^*(\Gamma)) = g^*(T_0', \dots, T_{m-1}', ff(T_{n-1})) = (T_0'', \dots, T_{k-1}'', gg(ff(T_n)))$$

and

$$(g \circ f)^*(\Gamma) = (T_0'', \dots, T_{k-1}'', (ff \hat{\circ} gg)(T_n))$$

we conclude that $g^*(f^*(\Gamma)) = (g \circ f)^*(\Gamma)$ by the composition of morphisms property of the functor \mathbf{LM} .

The second part of the seventh condition follows from the fact that Fl is fully faithful and from the second part of the seventh condition for $C(\mathbf{R})$.

This completes the verification of the conditions of [?, Definition 2.1] and the construction of the structure of a C0-system on $CC(\mathbf{R}, \mathbf{LM})$. To extend it to the structure of a C-system we apply [?, Proposition 2.4] and Lemma 5.11.

We provide the following lemma without a proof because the proof is immediate from the definitions and [?, Lemma 3.4].

Lemma 5.14 [2015.08.22.14] *The functor $Fl : CC(\mathbf{R}, \mathbf{LM}) \rightarrow C(\mathbf{R})$ is a homomorphism of C-systems.*

Remark 5.15 [2015.08.22.rem1] For $f = (\hat{m}, (\hat{n}, ff))$ we have

$$(lF(y)(f))^*(y_{n+1}) = (m+1, (\iota_0^0(y), \dots, \iota_0^{m-1}(y), ff(\iota_0^n(y))))$$

By (2) we have

$$ff(\iota_0^n(y)) = (\iota_0^n \hat{\circ} ff)(y) = \iota_0^m(y)$$

because both $\iota_0^n \widehat{\circ} f f$ and ι_0^m are elements of $R(0, m) = R(m)^{stn(0)}$ which is a one element set. Therefore

$$(lF(y)(f))^*(y_{n+1}) = y_{m+1}$$

The rest of the conditions that one needs to prove in order to show that lF is a homomorphism of C-systems is immediate from definitions and we obtain that

$$lF(y) : C(\mathbf{R}) \rightarrow CC(\mathbf{R}, \mathbf{LM})$$

is a homomorphism of C-systems. However it is not an isomorphism of C-systems and, as one can see for example from our study of sub-quotients of $CC(\mathbf{R}, lm)$, the C-systems $C(\mathbf{R})$ and $CC(\mathbf{R}, \mathbf{LM})$ can be very different. Cf. Remark ??.

Remark 5.16 [2015.08.22.rem2.from.old] There is another construction of a pre-category from $(\mathbf{R}, \mathbf{LM})$ which takes as an additional parameter a countable set Var (with decidable equality) which is called the set of variables. Let $F_n(Var)$ be the set of sequences of length n of pair-wise distinct elements of Var . Define the pre-category $CC(\mathbf{R}, \mathbf{LM}, Var)$ as follows. The set of objects of $CC(\mathbf{R}, \mathbf{LM}, Var)$ is

$$Ob(CC(\mathbf{R}, \mathbf{LM}, Var)) = \prod_{n \in \mathbf{N}} \prod_{(x_0, \dots, x_{n-1}) \in F_n(Var)} LM(\widehat{0}) \times \dots \times LM(\widehat{n})$$

For compatibility with the traditional type theory we will write the elements of $Ob(CC(\mathbf{R}, \mathbf{LM}, X))$ as sequences of the form $x_0 : E_1, \dots, x_{n-1} : E_{n-1}$. The set of morphisms is given by

$$Mor(CC(\mathbf{R}, \mathbf{LM}, Var)) = \prod_{\Gamma, \Gamma' \in Ob} R(l_f(\Gamma'), l_f(\Gamma))$$

The composition is defined in such a way that the projection

$$(x_0 : E_0, \dots, x_{n-1} : E_{n-1}) \mapsto (E_0, E_1, \dots, E_{n-1})$$

is a functor from $CC(\mathbf{R}, \mathbf{LM}, Var)$ to $CC(\mathbf{R}, \mathbf{LM})$.

This functor is clearly an equivalence but not an isomorphism of categories.

There are an obvious object pt , function ft and p -morphisms.

What is unclear is how to define operation f^* on objects such as to satisfy the first parts of the sixth and seventh conditions in the definition [?, Definition 2.1] of a C0-system. For $\Gamma' = (x'_0 : T'_0, \dots, x'_{m-1} : T'_{m-1})$, $\Gamma = (y_0 : T_0, \dots, y_n : T_n)$ and $f : \Gamma' \rightarrow ft(\Gamma)$ the object $f^*(\Gamma)$ must be of the form $(x'_0 : T'_0, \dots, x'_{m-1} : T'_{m-1}, z : T'_m)$ in order to satisfy the equation $ft(f^*(\Gamma)) = \Gamma'$ and we should have $z \neq x'_0, \dots, x'_{m-1}$.

Consider the case when $LM(\widehat{i}) = unit$ for all i . Then the problem is to construct functions

$$z_{n,m} : F_m(Var) \times F_{n+1}(Var) \times R(n, m) \rightarrow Var$$

such that $z_{n,m}(x', y, f)$ does not occur in x' and such that

$$\text{[2015.08.22.eq1b]} z_{n,m}((x_0, \dots, x_{n-1}), (x_0, \dots, x_{n-1}, x_n), \eta_n) = x_n \quad (32)$$

and for all $x' \in F_m$, $x'' \in F_k$, $y \in F_{n+1}$, $f \in R(n, m)$ and $g \in R(m, k)$ one has

$$\text{[2015.08.22.eq2b]} z_{n,k}(x'', y, g \widehat{\circ} f) = z_{m,k}(x'', (x'_0, \dots, x'_{m-1}, z_{m,n}(x', y, f)), g) \quad (33)$$

I do not know whether it is possible to construct a function z satisfying these two equations for a general R .

Lemma 5.17 [2015.08.22.15] *A morphism $f = (\Gamma', (\Gamma, ff))$, where $l_f(\Gamma) = n + 1$, belongs to $\widetilde{Ob}(CC(\mathbf{R}, \mathbf{LM}))$ if and only if $\Gamma' = ft(\Gamma)$ and $ff(i) = x_i^n$ for all $i = 0, \dots, n - 1$.*

Proof: It follows immediately from Lemma 2.8.

The following analog of Lemma 4.12 for the C-system $CC(\mathbf{R}, \mathbf{LM})$ provides us with the explicit form of the operation $f \mapsto s_f$.

Lemma 5.18 [2015.09.09.12] *Let $f = (\Gamma, (\Gamma', (f_0, \dots, f_{n-1})))$ where $n > 0$. Then one has*

$$[2015.09.09.eq1]_{s_f} = (\Gamma, ((ft(f))^*(\Gamma'), (x_0^m, \dots, x_{m-1}^m, f_{n-1}))) \quad (34)$$

where $ft(f) = f \circ p_{\Gamma'}$ and $m = l_f(\Gamma)$.

If $\Gamma = (m, (X_0, \dots, X_m))$ and $\Gamma' = (n, (Y_0, \dots, Y_m))$ then

$$[2015.09.09.eq2](ft(f))^*(\Gamma') = (X_0, \dots, X_{m-1}, (f_0, \dots, f_{n-2})(Y_{n-1})) \quad (35)$$

Proof: By definition s_f is a morphism from Γ to $(ft(f))^*(\Gamma)$. Therefore it is sufficient to show that the left hand side of (34) agrees with the right hand side after application of the homomorphism Fl . Since Fl is a homomorphism of C-systems, $Fl(s_f) = s_{Fl(s)}$ and our goal follows from Lemma 4.12.

Applying Lemma 5.8 we see that $ft(f) = (\Gamma, (ft(\Gamma'), (f_0, \dots, f_{n-2})))$. Therefore (35) follows immediately from (30).

For Lemma 5.20 below we first need some general results about homomorphisms of C-systems.

Lemma 5.19 [2015.09.03.12] *Let $F : CC \rightarrow CC'$ be a homomorphism of C-systems. Then one has:*

1. for $\Gamma \in CC$ and $i \in \mathbf{N}$ one has $F(p_{\Gamma,i}) = p_{F(\Gamma),i}$,
2. for $\Gamma, \Gamma' \in CC$, $\Gamma \leq \Gamma'$ implies $F(\Gamma) \leq F(\Gamma')$ and similarly for $<$,
3. for $\Gamma' \geq \Gamma$ and $f : \Delta \rightarrow \Gamma$ one has

$$F(f^*(\Gamma')) = (F(f))^*(F(\Gamma'))$$

4. for $\Gamma', \Gamma'' \geq \Gamma$, $a : \Gamma' \rightarrow \Gamma''$ over Γ and $f : \Delta \rightarrow \Gamma$ one has

$$F(f^*(a)) = (F(f))^*(F(a))$$

5. for Γ such that $l(\Gamma) > 0$ one has

$$F(\delta(\Gamma)) = \delta(F(\Gamma))$$

Proof: The proofs are straightforward and we leave them for the formalized version of the paper.

Lemma 5.20 [2015.09.03.11] For $i > 0$, $f : \Gamma' \rightarrow ft^i(\Gamma)$ and $s = (ft(\Gamma), (\Gamma, ss))$ in $\widetilde{Ob}(CC(\mathbf{R}, \mathbf{LM}))$ one has

$$f^*(s) = (f^*(ft(\Gamma)), (f^*(\Gamma), (x_0^{m+i-1}, \dots, x_{m+i-2}^{m+i-1}, (qq^{i-1}(ff))(ss(n-1))))))$$

where $m = l(\Gamma')$ and $n = l(\Gamma)$.

Proof: Since Fl is a full embedding, it is sufficient, in order to verify the equality of two morphisms to verify that their domain and codomain are equal and that their images under Fl are equal. For the domain and codomain it follows from the definition of f^* on morphisms. For the images under Fl it follows from the fact that Fl is a homomorphism of C-systems, Lemma 5.19(4) and Lemma 4.14.

Problem 5.21 [2015.08.22.prob1] To construct a bijection

$$[2009.10.15.eq2] mb_{\mathbf{R}, \mathbf{LM}} : \widetilde{Ob}(CC(\mathbf{R}, \mathbf{LM})) \rightarrow \prod_{n \in \mathbf{N}} Ob_{n+1}(\mathbf{R}, \mathbf{LM}) \times R(n) \quad (36)$$

Construction 5.22 [2015.08.22.constr1] [2014.06.30.12] Let $s \in \widetilde{Ob}(CC(\mathbf{R}, \mathbf{LM}))$. Then $s = (ft(\Gamma), (\Gamma, ss))$ for some $ss \in R(n, n+1)$ and $\Gamma = (n+1, X)$. We set:

$$mb_{\mathbf{R}, \mathbf{LM}}(s) = (n, (X, ss(n)))$$

To show that this is a bijection let us construct an inverse. For $n \in \mathbf{N}$, $X \in Ob_{n+1}(\mathbf{R}, \mathbf{LM})$ and $o \in R(n)$ let

$$mb_{\mathbf{R}, \mathbf{LM}}^{\dagger}(n, (X, o)) = (ft((n+1, X)), ((n+1, X), (x_0^n, \dots, x_{n-1}^n, o)))$$

This is a morphism from $ft(\Gamma)$ to Γ where $\Gamma = (n+1, X)$. The equation $mb_{\mathbf{R}, \mathbf{LM}}^{\dagger}(n, (\Gamma, o)) \circ p_{\Gamma} = Id_{ft(\Gamma)}$ follows from Lemma 5.17.

Let us show now that $mb_{\mathbf{R}, lm}$ and $mb_{\mathbf{R}, lm}^{\dagger}$ are mutually inverse bijections. Let $s \in \widetilde{Ob}(CC(\mathbf{R}, \mathbf{LM}))$ be as above, then:

$$mb_{\mathbf{R}, \mathbf{LM}}^{\dagger}(mb_{\mathbf{R}, \mathbf{LM}}(s)) = mb_{\mathbf{R}, \mathbf{LM}}^{\dagger}(n, (X, ss(n))) = (ft(\Gamma), (\Gamma, (x_0^n, \dots, x_{n-1}^n, ss(n)))) = s$$

where the last equality follows from the assumption that $s \in \widetilde{Ob}$ and Lemma 5.17.

On the other hand for $X \in Ob_{n+1}(\mathbf{R}, \mathbf{LM})$ and $o \in R(n)$ we have

$$mb_{\mathbf{R}, \mathbf{LM}}(mb_{\mathbf{R}, lm}^{\dagger}(n, (X, o))) = mb_{\mathbf{R}, \mathbf{LM}}(ft((n+1, X)), ((n+1, X), (x_0^n, \dots, x_{n-1}^n, o))) = (n, (X, o))$$

This completes Construction 5.22.

Lemma 5.23 [2015.09.09.13] Let $f = (\Gamma, (\Gamma', (f_0, \dots, f_{n-1})))$ where $\Gamma = (m, X)$, $\Gamma' = (n, Y)$ and $ff = (f_0, \dots, f_{n-1})$. Then one has

$$mb_{\mathbf{R}, \mathbf{LM}}(s_f) = (m, ((X_0, \dots, X_{m-1}, (f_0, \dots, f_{n-2})(Y_{n-1})), f_{n-1}))$$

Proof: It follows immediately from Lemma 5.18 and the formula for $mb_{\mathbf{R},\mathbf{LM}}$.

Consider operations $T', \tilde{T}', S', \tilde{S}'$ and δ' obtained by transport by means of the bijection of Construction 5.22 from the operations T, \tilde{T}, S and \tilde{S} and δ corresponding to the C-system $CC(\mathbf{R}, \mathbf{LM})$ (cf. Definition 4.18). Let us give an explicit description of these operations.

Theorem 5.24 [2015.08.26.th2] *Let $Ob = Ob(CC(\mathbf{R}, \mathbf{LM}))$ and let $\widetilde{Ob}' = \widetilde{Ob}'(\mathbf{R}, \mathbf{LM})$ be the right hand side of (36). One has:*

1. Operation T' is defined on the set T'_{dom} of pairs $(m, X), (n, Y) \in Ob$ such that $m > 0$, $n > m - 1$ and $X_i = Y_i$ for $i = 0, \dots, m - 2$. It takes values in Ob and is given by

$$T((m, X), (n, Y)) = (n + 1, (Y_0, \dots, Y_{m-2}, X_{m-1}, \partial_{m-1}^{m-1}(Y_{m-1}), \dots, \partial_{n-1}^{m-1}(Y_{n-1})))$$

2. Operation \tilde{T}' is defined on the set \tilde{T}'_{dom} of pairs $(m, X) \in Ob$, $(n, (Y, s)) \in \widetilde{Ob}'$ such that $m > 0$, $n + 1 > m - 1$ and $X_i = Y_i$ for $i = 0, \dots, m - 2$. It takes values in \widetilde{Ob}' and is given by

$$\tilde{T}'((m, X), (n, (Y, s))) = (n + 1, (T((m, X), (n, Y)), \partial_n^{m-1}(s)))$$

3. Operation S' is defined on the set of pairs $(m, (X, r)) \in \widetilde{Ob}'$, $(n, Y) \in Ob$ where $n > m + 1$ and $X_i = Y_i$ for $i = 0, \dots, m$. It takes values in the set Ob and is given by

$$S'((m, (X, r)), (n, Y)) = (n - 1, (Y_0, \dots, Y_{m-1}, \sigma_{\mathbf{LM}}(r, Y_{m+1}), \sigma_{\mathbf{LM}}(r, Y_{m+2}), \dots, \sigma_{\mathbf{LM}}(r, Y_{n-1})))$$

4. Operation \tilde{S}' is defined on the set of pairs $(m, (X, r)) \in \widetilde{Ob}'$, $(n, (Y, s)) \in \widetilde{Ob}'$ such that $n > m$ and $X_i = Y_i$ for $i = 0, \dots, m$. It takes values in \widetilde{Ob}' and is given by

$$\tilde{S}'((m, (X, r)), (n, (Y, s))) = (n - 1, (S'((m, (X, r)), (n + 1, Y)), \sigma_{\mathbf{R}}(r, s)))$$

5. Operation δ' is defined on the subset of (n, X) in Ob such that $n > 0$. It takes values in \widetilde{Ob}' and is given by

$$\delta'((n, X)) = (n, (p_{(n, X)}^*(n, X), x_{n-1}^n))$$

Proof: In the proof we will write mb and $mb^!$ instead of $mb_{\mathbf{R},\mathbf{LM}}$ and $mb_{\mathbf{R},lm}^!$. We have:

1. Operation T' is the same as operation T for $C(\mathbf{R})$ since \widetilde{Ob} is not involved in it. The form of T'_{dom} is obtained by unfolding definitions and using Lemma 5.8(1).

The operation itself is given by

$$T'((m, X), (n, Y)) = p_{(m, X)}^*((n, Y)) = (m, (Y_0, \dots, Y_{m-1}, \partial_{m-1}^{m-1}(Y_{m-1}), \dots, \partial_{n-1}^{m-1}(Y_{n-1})))$$

where the first equality is by Definition 4.18(1) and the second by Lemma 5.10.

2. Operation \tilde{T}' is defined on the set of pairs $((m, X) \in Ob, (n, (Y, s)) \in \widetilde{Ob}'$) such that $m > 0$ and $\partial(mb^1(n, (Y, s))) > ft(m, X)$. Since $\partial(mb^1(n, (Y, s))) = (n + 1, Y)$ we obtain the required domain of definition from Lemma 5.8(1).

To verify the formula for the operation itself consider the equalities:

$$\begin{aligned} \tilde{T}'((m, X), (n, (Y, s))) &= mb(p_{(m, X)}^*(mb^1(n, (Y, s)))) = \\ &mb(p_{(m, X)}^*(ft((n + 1, Y)), ((n + 1, Y), (x_0^n, \dots, x_{n-1}^n, s)))) \end{aligned}$$

where the first equality is by Definition 4.18(2). By Lemma 5.20 we can extend these equalities as follows:

$$\begin{aligned} &mb(p_{(m, X)}^*(ft((n + 1, Y)), ((n + 1, Y), (x_0^n, \dots, x_{n-1}^n, s)))) = \\ &mb(p_{\Gamma'}^*(ft(\Gamma)), (p_{\Gamma'}^*(\Gamma), (x_0^{n+1}, \dots, x_n^{n+1}, (qq^{n-m+1}(l_{m-1}^1)(s)))) = \\ &(n + 1, (p_{\Gamma'}^*(\Gamma), \partial_n^{m-1}(s))) = (n + 1, (T((m, X), (n + 1, Y)), \partial_n^{m-1}(s))) \end{aligned}$$

where $\Gamma' = (m, X)$, $\Gamma = (n + 1, Y)$, the first equality is by Lemma 5.20, the second by Lemma 2.10 and the third by Definition 4.18(1).

3. Operation S' is defined on the set of pairs $((m, (X, r)) \in \widetilde{Ob}'$, $(n, Y) \in Ob$) such that $(n, Y) > \partial(mb^1(m, (X, r)))$. Since $\partial(mb^1(m, (X, r))) = (m + 1, X)$ we obtained the required domain of definition. The operation itself is given by

$$\begin{aligned} S'((m, (X, r)), (n, Y)) &= (mb^1((m, (X, r))))^*((n, Y)) = \\ &(ft((m + 1, X)), ((m + 1, X), (x_0^m, \dots, x_{m-1}^m, r)))^*((n, Y)) = \\ &(ft((m + 1, X)), ((m + 1, X), (x_0^m, \dots, x_{m-1}^m, r)))^*((n, Y), i) \end{aligned}$$

where $i = n - m - 1$. By Lemma 5.9 we can extend this equality as follows:

$$\begin{aligned} &(n - 1, (X_0, \dots, X_{m-1}, rr(Y_{m+1}), qq(rr)(Y_{m+2}), \dots, qq^{n-m-2}(rr)(Y_{n-1}))) = \\ &(n - 1, (Y_0, \dots, Y_{m-1}, rr(Y_{m+1}), qq(rr)(Y_{m+2}), \dots, qq^{n-m-2}(rr)(Y_{n-1}))) \end{aligned}$$

where $rr = (x_0^m, \dots, x_{m-1}^m, r)$ and the second equality holds by our assumption on X and Y . The required formula follows from the equality

$$qq^j(rr)(Y_{m+j+1}) = \sigma_{\mathbf{LM}}(r, Y_{m+j+1})$$

4. Operation \tilde{S}' is defined on the set of pairs $(m, (X, r)) \in \widetilde{Ob}'$, $(n, (Y, s)) \in \widetilde{Ob}'$ such that

$$\partial(mb^1((n, (Y, s)))) > \partial(mb^1(m, (X, r)))$$

which is equivalent to

$$(n + 1, Y) > (m + 1, X)$$

which is, in turn, equivalent to the conditions in the theorem. The operation itself is given by:

$$\begin{aligned} \tilde{S}'((m, (X, r)), (n, (Y, s))) &= mb((mb^1(m, (X, r)))^*(mb^1((n, (Y, s))))) = \\ &mb((ft((m + 1, X)), ((m + 1, X), rr))^*(ft(n + 1, Y), ((n + 1, Y), ss))) \end{aligned}$$

where

$$\begin{aligned} rr &= (x_0^m, \dots, x_{m-1}^n, r) \\ ss &= (x_0^n, \dots, x_{n-1}^n, s) \end{aligned}$$

By Lemma 5.20 we can extend this equality as follows:

$$\begin{aligned} &mb((ft((m+1, X)), ((m+1, X), rr))^*(ft(n+1, Y), ((n+1, Y), ss))) = \\ &mb(((mb^1(m, (X, r)))^*(ft(\Gamma)), ((mb^1(m, (X, r)))^*(\Gamma), (x_0^{n-1}, \dots, x_{n-2}^{n-1}, (qq^{n-m-1}(rr))(s)))))) = \\ &(n-1, ((mb^1(m, (X, r)))^*(\Gamma), (qq^{n-m-1}(rr))(s))) = \\ &(n-1, (S'((m, (X, r)), (n+1, Y))), \sigma_{\mathbf{R}}(r, s)) \end{aligned}$$

where $\Gamma' = ft(m, X)$, $\Gamma = (n+1, Y)$, the first equality is by Lemma 2015.09.03.11 and the third equality by Definition Definition 4.18(3).

5. Operation δ' is defined on the subset $(n, X) \in Ob$ such that $n > 0$ and is given by

$$\delta'((n, X)) = mb(\delta((n, X)))$$

Therefore it is sufficient to show that

$$\delta((n, X)) = ((n, X), (p_{(n, X)}^*((n, X)), (x_0^n, \dots, x_{n-1}^n, x_{n-1}^n)))$$

By Definition 4.18(5), $\delta((n, X))$ is a morphism from (n, X) to $p_{(n, X)}^*((n, X))$. Therefore, since Fl is a fully faithful functor it is sufficient to show that

$$Fl(\delta((n, X))) = (n, (n+1, (x_0^n, \dots, x_{n-1}^n, x_{n-1}^n)))$$

which follows from Lemma 5.19(5) and Lemma 4.15.

Remark 5.25 [2015.09.07.rem5] As we did in Remark 4.20 for R so we can do for LM , representing the category of pairs (R, LM) , which is equivalent to the full subcategory of finitary objects in the Hirschowitz-Maggesi category for *Sets*, as follows.

Given an l-algebraic theory $l\mathbf{R} = (lR, l, \eta, \partial, \sigma)$ we can define a left l-module $l\mathbf{LM}$ over \mathbf{R} as a quadruple:

1. a set lLM ,
2. a function $l_{\mathbf{LM}} : lLM \rightarrow \mathbf{N}$,
3. a function $\partial_{\mathbf{LM}} : \{E \in lLM, i \in \mathbf{N} \mid l_{\mathbf{LM}}(E) \geq i\} \rightarrow lLM$,
4. a function

$$\sigma_{\mathbf{LM}} : \{r \in lR, E \in lLM \mid l_{\mathbf{LM}}(E) > l_{\mathbf{R}}(r)\} \rightarrow lLM$$

where operations $l_{\mathbf{LM}}$, $\partial_{\mathbf{LM}}$ and $\sigma_{\mathbf{LM}}$ satisfy some conditions.

Once these conditions are properly established the category of such pairs $(l\mathbf{R}, l\mathbf{LM})$ should be equivalent to the Hirschowitz-Maggesi category and in particular the systems of expressions associated with binding signatures can be described as universal objects carrying some additional operations in this category.

Let us emphasize again that the main reason to consider these l-versions of the main constructions of this paper is that they should be easier to formalize in systems without dependent types such as HOL or ZF.

6 C-subsystems of $CC(\mathbf{R}, \mathbf{LM})$.

Let CC be a C-subsystem of $CC(\mathbf{R}, \mathbf{LM})$. Then, obviously, $Ob(CC) \subset Ob(CC(\mathbf{R}, \mathbf{LM}))$ and $\widetilde{Ob}(CC) \subset \widetilde{Ob}(CC(\mathbf{R}, \mathbf{LM}))$. It was proved in [?, Lemma 4.1] that the function

$$CC \mapsto (Ob(CC) \subset Ob(CC(\mathbf{R}, \mathbf{LM})), \widetilde{Ob}(CC) \subset \widetilde{Ob}(CC(\mathbf{R}, \mathbf{LM})))$$

is an injection from the set of C-subsystems of $CC(\mathbf{R}, \mathbf{LM})$ to the set of pairs of subsets in $Ob(CC(\mathbf{R}, \mathbf{LM}))$ and $\widetilde{Ob}(CC(\mathbf{R}, \mathbf{LM}))$ and in [?, Proposition 4.3] that its image consists of those pairs of subsets (B, \widetilde{B}) that satisfy the following conditions:

1. $pt \in B$
2. if $X \in B$ then $ft(X)$ is in B ,
3. if $r \in \widetilde{B}$ then $\partial(r) \in \widetilde{B}$,
4. if $X \in B$, $s \in \widetilde{B}$ and $(X, s) \in \widetilde{T}_{dom}$ then $\widetilde{T}(X, s) \in \widetilde{B}$,
5. if $r, s \in \widetilde{B}$ and $(r, s) \in \widetilde{S}_{dom}$ then $\widetilde{S}(s, r) \in \widetilde{B}$,
6. if $X \in B$ and $X \in \delta_{dom}$ then $\delta(X) \in \widetilde{B}$.

Clearly the same result holds for subsets $B \subset Ob(CC(\mathbf{R}, \mathbf{LM}))$ and $\widetilde{B}' \subset \widetilde{Ob}'$ if we replace ∂ by ∂' (where $\partial'(m, (X, r)) = X$), T by T' and \widetilde{S} by \widetilde{S}' .

Let us combine this result with the explicit description of the operations T' and \widetilde{S}' obtained in Theorem 5.24 and the following notation:

1. for (m, X) in $Ob(CC(\mathbf{R}, \mathbf{LM}))$ we write $(X \triangleright^B)$ if $(m, X) \in B$, when no confusion is possible we will omit the superscript \triangleright^B from \triangleright ,
2. for $(m, (X, r))$ in $\widetilde{Ob}'(\mathbf{R}, \mathbf{LM})$ we write $(X \triangleright^{\widetilde{B}'} r : X_m)$ if $(m, (X, r))$ is in \widetilde{B}' , similarly to what was said above, if no confusion is possible we will omit the superscript \widetilde{B}' from \triangleright .

Those who are used to the turnstile notations $(X_0, \dots, X_{m-1} \vdash r : X_m)$ for the statement that the “judgement” that r is an object of type X_m in the context (X_0, \dots, X_{m-1}) is derivable, should only replace \vdash with \triangleright and sometimes add the superscript with B or \widetilde{B} referring to the class of judgements that are derivable relative to the particular systems of inference rules under consideration⁶. In a subsequent paper we will consider the case when the sets B and \widetilde{B} are the sets of derivable judgements relative to a set of inference rules with both the concept of an inference rule and that of the derivability being precisely defined in the mathematical terms. So far, B and \widetilde{B} may refer to any subsets on Ob and \widetilde{Ob} respectively.

Another group of notations that we would like to introduce is as follows. Recall that our use of sequences such as (X_0, \dots, X_{m-1}) is a notational convention and that the object underlying this notation is a dependent function with the domain $stn(m)$ and the codomain function $i \mapsto R(i)$. In what follows we will use more such notations explained below:

⁶The reason for the replacement of the familiar \vdash with the unfamiliar \triangleright is that \vdash carries a meaning in logic that does not always agree with its use in type theory, for example $\Gamma \vdash$ reads as Γ implies the empty set set of conclusions which always holds and so $\Gamma \vdash$ always holds, while the type theoretic meaning is that Γ is a well-formed context of the type theory under considerations.

1. Given a dependent function on $stn(m)$, $\Gamma = (X_0, \dots, X_{m-1})$ where $X_i \in LM(\hat{i})$ and a dependent function on $\{j \in \mathbf{N} \mid j \geq m, j < n\}$, $\Delta = (X_m, \dots, X_{n-1})$ where $X_j \in LM(\hat{j})$ we will write (Γ, Δ) for the dependent function of the obvious form on $stn(n)$,

2. Given $\Delta = (X_m, \dots, X_{m+j})$ where $X_i \in LM(\hat{i})$ and $k \geq i$ we define

$$\partial^k(\Delta) = (\partial_m^k(X_m), \dots, \partial_{m+j}^k(X_{m+j})).$$

3. Given $\Delta = (X_m, \dots, X_{m+j})$ where $X_i \in LM(\hat{i})$ and $r \in R(j)$ where $j < m$ we define

$$\sigma_{\mathbf{LM}}(s, \Delta) = (\sigma_{\mathbf{LM}}(s, X_m), \dots, \sigma_{\mathbf{LM}}(s, X_{m+j})).$$

In this notation the description of the operations $T, \tilde{T}, S, \tilde{S}$ and δ given in Theorem 5.24 takes the form:

1. $T((\Gamma, X_m \triangleright), (\Gamma, \Delta \triangleright)) = (\Gamma, X_m, \partial^m(\Delta) \triangleright)$ where $m = l_f(\Gamma)$,
2. $\tilde{T}((\Gamma, X_m \triangleright), (\Gamma, \Delta \triangleright s : Y_n)) = (\Gamma, X_m, \partial^m(\Delta) \triangleright \partial^m(s : Y_n))$ where $m = l_f(\Gamma)$,
3. $S((\Gamma \triangleright r : X_m), (\Gamma, X_m, \Delta \triangleright)) = (\Gamma, \sigma_{\mathbf{LM}}(r, \Delta) \triangleright)$,
4. $\tilde{S}((\Gamma \triangleright r : X_m), (\Gamma, X_m, \Delta \triangleright s : Y_n)) = (\Gamma, \sigma_{\mathbf{LM}}(r, \Delta) \triangleright \sigma_{\mathbf{LM}}(r, s) : \sigma_{\mathbf{LM}}(r, Y_n))$,
5. $\delta(\Gamma, X_m \triangleright) = (\Gamma, X_m \triangleright x_m^{m+1} : X_m)$ where $m = l_f(\Gamma)$.

The following result is an immediate corollary of [?, Proposition 4.3] together with the description of the operations $T, \tilde{T}, S, \tilde{S}$ and δ for $CC(\mathbf{R}, \mathbf{LM})$ which is given above.

Proposition 6.1 [2009.10.16.prop3] *Let $(\mathbf{R}, \mathbf{LM})$ be as above. Let (B, \tilde{B}') be a pair of subsets in $Ob(CC(\mathbf{R}, \mathbf{LM}))$ and $\tilde{Ob}'(\mathbf{R}, \mathbf{LM})$ respectively. Then the following two conditions are equivalent:*

1. *There exists a C-subsystem CC of $CC(\mathbf{R}, \mathbf{LM})$ such that $Ob(CC) = B$ and $mb_{\mathbf{R}, lm}(\tilde{Ob}(CC)) = \tilde{B}'$ as subsets in $Ob(CC(\mathbf{R}, \mathbf{LM}))$ and $\tilde{Ob}(\mathbf{R}, \mathbf{LM})$ respectively.*
2. *The following conditions hold:*
 - (a) (\triangleright^B)
 - (b) $(\Gamma, X_m \triangleright^B) \Rightarrow (\Gamma \triangleright^B)$
 - (c) $(\Gamma \triangleright^{\tilde{B}} r : X_m) \Rightarrow (\Gamma, X_m \triangleright^B)$
 - (d) $(\Gamma, X_m \triangleright^B) \wedge (\Gamma, \Delta, \triangleright^{\tilde{B}} s : Y_n) \Rightarrow (\Gamma, X_m, \partial^m(\Delta) \triangleright^{\tilde{B}} \partial^m(s) : \partial^m(Y_n))$ where $m = l_f(\Gamma)$
 - (e) $(\Gamma \triangleright^{\tilde{B}} r : X_m) \wedge (\Gamma, X_m, \Delta \triangleright^{\tilde{B}} s : Y_n) \Rightarrow (\Gamma, \sigma_{\mathbf{LM}}(r, \Delta) \triangleright^{\tilde{B}} \sigma_{\mathbf{LM}}(r, s) : \sigma_{\mathbf{LM}}(r, Y_n))$,
 - (f) $(\Gamma, X_m \triangleright^B) \Rightarrow (\Gamma, X_m \triangleright^{\tilde{B}} x_m^{m+1} : X_m)$ where $m = l_f(\Gamma)$.

In addition if a C-subsystem CC satisfying the conditions of the proposition exists then it is unique.

Note that conditions (4) and (5) together with condition (6) and condition (3) imply the following

$$\mathbf{4a} \quad (\Gamma, X_m \triangleright^B) \wedge (\Gamma, \Delta \triangleright^B) \Rightarrow (\Gamma, X_m, \partial^m(\Delta) \triangleright^B) \text{ where } m = l_f(\Gamma),$$

5a $(\Gamma \triangleright^{\tilde{B}} r : X_m) \wedge (\Gamma, X_m, \Delta \triangleright^B) \Rightarrow (\Gamma, \sigma_{\mathbf{LM}}(r, \Delta) \triangleright^B).$

Remark 6.2 [2010.08.07.rem1] If one re-writes the conditions of Proposition 6.1 in the more familiar in type theory form where the variables introduced in the context are named rather than directly numbered and where the numbering is not from 0 but from 1 one arrives at the following rules:

$$\frac{}{\triangleright^B} \quad \frac{x_1 : T_1, \dots, x_n : T_n \triangleright^B}{x_1 : T_1, \dots, x_{n-1} : T_{n-1} \triangleright^B} \quad \frac{x_1 : T_1, \dots, x_n : T_n \triangleright^{\tilde{B}'} t : T}{x_1 : T_1, \dots, x_n : T_n, y : T \triangleright^B}$$

$$\frac{x_1 : T_1, \dots, x_n : T_n, y : T \triangleright^B \quad x_1 : T_1, \dots, x_n : T_n, \dots, x_m : T_m \triangleright^{\tilde{B}'} r : R}{x_1 : T_1, \dots, x_n : T_n, y : T, x_{n+1} : T_{n+1}, \dots, x_m : T_m \triangleright^{\tilde{B}'} r : R}$$

$$\frac{x_1 : T_1, \dots, x_n : T_n \triangleright^{\tilde{B}'} s : S \quad x_1 : T_1, \dots, x_n : T_n, y : S, x_{n+1} : T_{n+1}, \dots, x_m : T_m \triangleright^{\tilde{B}'} r : R}{x_1 : T_1, \dots, x_n : T_n, x_{n+1} : T_{n+1}[s/y], \dots, x_m : T_m[s/y] \triangleright^{\tilde{B}'} (r : R)[s/y]}$$

$$\frac{x_1 : E_1, \dots, x_n : E_n \triangleright^B}{x_1 : E_1, \dots, x_n : E_n \triangleright^{\tilde{B}'} x_n : E_n}$$

which are similar (and probably equivalent) to the “basic rules of DTT” given in [?, p.585]. The advantage of the rules given here is that they are precisely the ones which are necessary and sufficient for a given collection of contexts and judgements to define a C-subsystem.

The following lemma is needed in order to show that the pre-category that we obtain from (B, \tilde{B}) using our construction of a sub-C-system from a pair of subsets in Ob and \tilde{Ob}' satisfying certain conditions has the same morphisms as the pre-category whose construction from such subsets is outlined in more traditional texts such as [?]. The fact that one obtains the same composition is more difficult to prove since I do not know of any traditional texts where the composition would be described in sufficient detail.

Lemma 6.3 [2009.11.05.11] *Let (B, \tilde{B}') be a pair of subsets satisfying the conditions of Proposition 6.1 and let CC be the corresponding C-subsystem of $CC(\mathbf{R}, \mathbf{LM})$. Let $(m, X), (n, Y) \in B$ and*

$$f = ((m, X), ((n, Y), (f_0, \dots, f_{n-1}))) \in Hom_{CC(\mathbf{R}, \mathbf{LM})}((m, X), (n, Y))$$

Then $f \in Hom_{CC}((m, X), (n, Y))$ if and only if

$$f \circ p_{(n, Y)} \in Hom_{CC}((m, X), ft((n, Y)))$$

and

$$\text{[2015.09.09.eq4]}(X_0, \dots, X_{m-1} \triangleright^{\tilde{B}'} f_{n-1} : (f_0, \dots, f_{n-2})(Y_{n-1})) \quad (37)$$

Proof: The expression (37) is our notation for the statement that

$$(m, ((X_0, \dots, X_{m-1}, (f_0, \dots, f_{n-2})(Y_{n-1})), f_{n-1})) \in \tilde{B}'$$

By Lemma 5.23 this is equivalent to the condition that $s_f \in \tilde{B}$ where $\tilde{B} = mb_{\mathbf{R}, \mathbf{LM}}^!(\tilde{B}')$. By definition of CC in Proposition 6.1 we have $\tilde{B} = \tilde{Ob}(CC)$.

By [?, Definition 2.3(3)] we have

$$f = s_f \circ q(f \circ p_{(n, Y)}, (n, Y))$$

therefore if $s_f \in Mor(CC)$ and $f \circ p_{(n, Y)} \in Mor(CC)$ then $f \in Mor(CC)$ since CC is a C-subsystem of $CC(\mathbf{R}, \mathbf{LM})$. This proves the "if" part of the Lemma.

The "only if" part follows similarly from the assumption that CC is a C-subsystem and therefore both $f \circ p_{(n, Y)}$ and s_f belong to $Mor(CC)$.

7 Operations σ and $\tilde{\sigma}$.

C-systems of the form $CC(\mathbf{R}, \mathbf{LM})$ have an important additional structure which will play a role in the next section. This structure is given by two operations:

1. For $(m, X), (n, Y)$ where $m \geq n$ we define $\sigma((m, X), (n, Y))$ as the object given in the sequence notation by

$$\sigma((m, X), (n, Y)) = (Y_0, \dots, Y_{n-1}, X_n, \dots, X_{m-1})$$

This gives us an operation with values in Ob defined on the subset of $Ob \times Ob$ which consists of pairs (Γ, Γ') such that $l_f(\Gamma) \geq l_f(\Gamma')$. Note that for $m = n$ one has

$$\sigma((m, X), (n, Y)) = (n, Y)$$

2. for $(m, (X, r))$ in \tilde{Ob}' and $(n, Y) \in Ob$ such that $m + 1 \geq n$

$$\tilde{\sigma}((m, (X, r)), (n, Y)) = (m, (\sigma((m + 1, X), (n, Y)), r))$$

This gives us an operation with values in \tilde{Ob}' defined on the subset of $\tilde{Ob}' \times Ob$ which consists of pairs (r, Γ') such that $l_f(\partial(r)) \leq l_f(\Gamma')$.

Both operations can be easily re-written using \tilde{Ob} instead of \tilde{Ob}' but no analog of these operations exist on general, non-split, C-systems (cf. Remark ??).

8 Regular sub-quotients of $CC(\mathbf{R}, \mathbf{LM})$.

The concept of a regular congruence relation on a C-system was introduced in [?, Section 5]. This concept is crucial for the mathematics analysis of quotients of the C-systems that arise from "definitional equalities" of type systems.

We are going here to describe the regular congruence relations on C-systems of the form $CC(\mathbf{R}, \mathbf{LM})$ in terms of pairs of subsets

$$Beq \subset \coprod_{n \in \mathbf{N}} Ob_n(\mathbf{R}, \mathbf{LM}) \times (LM(\hat{n}) \times LM(\hat{n}))$$

$$\widetilde{Beq}' \subset \coprod_{n \in \mathbf{N}} Ob_n(\mathbf{R}, \mathbf{LM}) \times (LM(\widehat{n}) \times (R(n) \times R(n)))$$

satisfying certain conditions.

For $\Gamma \in Ob(CC(\mathbf{R}, \mathbf{LM}))$ and $S_1, S_2 \in LM(n)$ we write $(\Gamma \triangleright^{Beq} S_1 = S_2)$ to signify that $(\Gamma, (S_1, S_2)) \in Beq$. Similarly for $T \in LM(n)$ and $o, o' \in R(n)$ we write $(\Gamma \triangleright^{\widetilde{Beq}'} o = o' : T)$ to signify that $(T_1, \dots, T_n, S, o, o') \in \widetilde{Beq}'$. When no confusion is possible we will omit the subscripts Beq and \widetilde{Beq}' at \triangleright .

Similarly we will write \triangleright instead of \triangleright^C and \triangleright instead of $\triangleright^{\widetilde{B}}$ if the subsets C and \widetilde{B} are unambiguously determined by the context.

Definition 8.1 [*simandsimeq*] Given subsets $C, \widetilde{B}, Beq, \widetilde{Beq}$ as above define relations \sim on C and \simeq on \widetilde{B} as follows:

1. for $\Gamma = (T_1, \dots, T_n), \Gamma' = (T'_1, \dots, T'_n)$ in C we set $\Gamma \sim \Gamma'$ iff $ft(\Gamma) \sim ft(\Gamma')$ and

$$T_1, \dots, T_{n-1} \triangleright T_n = T'_n,$$

2. for $(\Gamma \triangleright o : S), (\Gamma' \triangleright o' : S')$ in \widetilde{B} we set $(\Gamma \triangleright o : S) \simeq (\Gamma' \triangleright o' : S')$ iff $(\Gamma, S) \sim (\Gamma', S')$ and

$$(\Gamma \triangleright o = o' : S).$$

Proposition 8.2 [*2014.07.10.prop1*] Let $C, \widetilde{B}, Beq, \widetilde{Beq}$ be as above and suppose in addition that one has:

1. C and \widetilde{B} satisfy conditions (1)-(6) of Proposition 6.1 which are referred to below as conditions (1.1)-(1.6) of the present proposition,

2.

- (a) $(\Gamma \triangleright T = T') \Rightarrow (\Gamma, T \triangleright)$
- (b) $(\Gamma, T \triangleright) \Rightarrow (\Gamma \triangleright T = T)$
- (c) $(\Gamma \triangleright T = T') \Rightarrow (\Gamma \triangleright T' = T)$
- (d) $(\Gamma \triangleright T = T') \wedge (\Gamma \triangleright T' = T'') \Rightarrow (\Gamma \triangleright T = T'')$

3.

- (a) $(\Gamma \triangleright o = o' : T) \Rightarrow (\Gamma \triangleright o : T)$
- (b) $(\Gamma \triangleright o : T) \Rightarrow (\Gamma \triangleright o = o : T)$
- (c) $(\Gamma \triangleright o = o' : T) \Rightarrow (\Gamma \triangleright o' = o : T)$
- (d) $(\Gamma \triangleright o = o' : T) \wedge (\Gamma \triangleright o' = o'' : T) \Rightarrow (\Gamma \triangleright o = o'' : T)$

4.

- (a) $(\Gamma_1 \triangleright T = T') \wedge (\Gamma_1, T, \Gamma_2 \triangleright S = S') \Rightarrow (\Gamma_1, T', \Gamma_2 \triangleright S = S')$
- (b) $(\Gamma_1 \triangleright T = T') \wedge (\Gamma_1, T, \Gamma_2 \triangleright o = o' : S) \Rightarrow (\Gamma_1, T', \Gamma_2 \triangleright o = o' : S)$
- (c) $(\Gamma \triangleright S = S') \wedge (\Gamma \triangleright o = o' : S) \Rightarrow (\Gamma \triangleright o = o' : S')$

5.

- (a) $(\Gamma_1, T \triangleright) \wedge (\Gamma_1, \Gamma_2 \triangleright S = S') \Rightarrow (\Gamma_1, T, t_{i+1}\Gamma_2 \triangleright t_{i+1}S = t_{i+1}S')$ $i = l(\Gamma)$
- (b) $(\Gamma_1, T \triangleright) \wedge (\Gamma_1, \Gamma_2 \triangleright o = o' : S) \Rightarrow (\Gamma_1, T, t_{i+1}\Gamma_2 \triangleright t_{i+1}o = t_{i+1}o' : t_{i+1}S)$ $i = l_f(\Gamma)$

6.

$$\begin{aligned}
& (a) \quad (\Gamma_1, T, \Gamma_2 \triangleright S = S') \wedge (\Gamma_1 \triangleright r : T) \Rightarrow \\
& (\Gamma_1, s_{i+1}(\Gamma_2[r/i + 1]) \triangleright s_{i+1}(S[r/i + 1]) = s_{i+1}(S'[r/i + 1])) \quad i = l_f(\Gamma_1) \\
& (b) \quad (\Gamma_1, T, \Gamma_2 \triangleright o = o' : S) \wedge (\Gamma_1 \triangleright r : T) \Rightarrow \\
& (\Gamma_1, s_{i+1}(\Gamma_2[r/i + 1]) \triangleright s_{i+1}(o[r/i + 1]) = s_{i+1}(o'[r/i + 1]) : s_{i+1}(S[r/i + 1])) \quad i = l_f(\Gamma_1)
\end{aligned}$$

7.

$$\begin{aligned}
& (a) \quad (\Gamma_1, T, \Gamma_2, S \triangleright) \wedge (\Gamma_1 \triangleright r = r' : T) \Rightarrow \\
& (\Gamma_1, s_{i+1}(\Gamma_2[r/i + 1]) \triangleright s_{i+1}(S[r/i + 1]) = s_{i+1}(S'[r'/i + 1])) \quad i = l_f(\Gamma_1) \\
& (b) \quad (\Gamma_1, T, \Gamma_2 \triangleright o : S) \wedge (\Gamma_1 \triangleright r = r' : T) \Rightarrow \\
& (\Gamma_1, s_{i+1}(\Gamma_2[r/i + 1]) \triangleright s_{i+1}(o[r/i + 1]) = s_{i+1}(o'[r'/i + 1]) : s_{i+1}(S[r/i + 1])) \quad i = l_f(\Gamma_1)
\end{aligned}$$

Then the relations \sim and \simeq are equivalence relations on C and \tilde{B} which satisfy the conditions of [?, Proposition 5.4] and therefore they correspond to a regular congruence relation on the C -system defined by (C, \tilde{B}) .

Lemma 8.3 [iseqrelsimpl1] *One has:*

1. If conditions (1.2), (4a) of the proposition hold then $(\Gamma \triangleright S = S') \wedge (\Gamma \sim \Gamma') \Rightarrow (\Gamma' \triangleright S = S')$.
2. If conditions (1.2), (1.3), (4a), (4b), (4c) hold then $(\Gamma \triangleright o = o' : S) \wedge ((\Gamma, S) \sim (\Gamma', S')) \Rightarrow (\Gamma' \triangleright o = o' : S')$.

Proof: By induction on $n = l_f(\Gamma) = l_f(\Gamma')$.

(1) For $n = 0$ the assertion is obvious. Therefore by induction we may assume that $(\Gamma \triangleright S = S') \wedge (\Gamma \sim \Gamma') \Rightarrow (\Gamma' \triangleright S = S')$ for all $i < n$ and all appropriate Γ, Γ', S and S' and that $(T_1, \dots, T_n \triangleright S = S') \wedge (T_1, \dots, T_n \sim T'_1, \dots, T'_n)$ holds and we need to show that $(T'_1, \dots, T'_n \triangleright S = S')$ holds. Let us show by induction on j that $(T'_1, \dots, T'_j, T_{j+1}, \dots, T_n \triangleright S = S')$ for all $j = 0, \dots, n$. For $j = 0$ it is a part of our assumptions. By induction we may assume that $(T'_1, \dots, T'_j, T_{j+1}, \dots, T_n \triangleright S = S')$. By definition of \sim we have $(T_1, \dots, T_j \triangleright T_{j+1} = T'_{j+1})$. By the inductive assumption we have $(T'_1, \dots, T'_j \triangleright T_{j+1} = T'_{j+1})$. Applying (4a) with $\Gamma_1 = (T'_1, \dots, T'_j)$, $T = T_{j+1}$, $T' = T'_{j+1}$ and $\Gamma_2 = (T_{j+2}, \dots, T_n)$ we conclude that $(T'_1, \dots, T'_{j+1}, T_{j+2}, \dots, T_n \triangleright S = S')$.

(2) By the first part of the lemma we have $\Gamma' \triangleright S = S'$. Therefore by (4c) it is sufficient to show that $(\Gamma \triangleright o = o' : S) \wedge (\Gamma \sim \Gamma') \Rightarrow (\Gamma' \triangleright o = o' : S)$. The proof of this fact is similar to the proof of the first part of the lemma using (4b) instead of (4a).

Lemma 8.4 [iseqrelsim] *One has:*

1. Assume that conditions (1.2), (2b), (2c), (2d) and (4a) hold. Then \sim is an equivalence relation.
2. Assume that conditions of the previous part of the lemma as well as conditions (1.3), (3b), (3c), (3d), (4b) and (4c) hold. Then \simeq is an equivalence relation.

Proof: By induction on $n = l_f(\Gamma) = l_f(\Gamma')$.

(1) Reflexivity follows directly from (1.2) and (2b). For $n = 0$ the symmetry is obvious. Let $(\Gamma, T) \sim (\Gamma', T')$. By induction we may assume that $\Gamma' \sim \Gamma$. By Lemma 8.3(a) we have $(\Gamma' \triangleright T = T')$ and by (2c) we have $(\Gamma' \triangleright T' = T)$. We conclude that $(\Gamma', T') \sim (\Gamma, T)$. The proof of transitivity is by a similar induction.

(2) Reflexivity follows directly from reflexivity of \sim , (1.3) and (3b). Symmetry and transitivity are also easy using Lemma 8.3.

From this point on we assume that all conditions of Proposition 8.2 hold. Let $C' = C / \sim$ and $\tilde{B}' = \tilde{B} / \simeq$. It follows immediately from our definitions that the functions $ft : C \rightarrow C$ and $\partial : \tilde{B} \rightarrow C$ define functions $ft' : C' \rightarrow C'$ and $\partial' : \tilde{B}' \rightarrow C'$.

Lemma 8.5 [surjl1] *The conditions (3) and (4) of [?, Proposition 5.4] hold for \sim and \simeq .*

Proof: 1. We need to show that for $(\Gamma, T \triangleright)$, and $\Gamma \sim \Gamma'$ there exists $(\Gamma', T' \triangleright)$ such that $(\Gamma, T) \sim (\Gamma', T')$. It is sufficient to take $T = T'$. Indeed by (2b) we have $\Gamma \triangleright T = T$, by Lemma 8.3(1) we conclude that $\Gamma' \triangleright T = T$ and by (1a) that $\Gamma', T \triangleright$.

2. We need to show that for $(\Gamma \triangleright o : S)$ and $(\Gamma, S) \sim (\Gamma', S')$ there exists $(\Gamma' \triangleright o' : S')$ such that $(\Gamma' \triangleright o' : S') \simeq (\Gamma \triangleright o : S)$. It is sufficient to take $o' = o$. Indeed, by (3b) we have $(\Gamma \triangleright o = o : S)$, by Lemma 8.3(2) we conclude that $(\Gamma' \triangleright o = o : S')$ and by (2a) that $(\Gamma' \triangleright o : S')$.

Lemma 8.6 [TSetc] *The equivalence relations \sim and \simeq are compatible with the operations $T, \tilde{T}, S, \tilde{S}$ and δ .*

Proof: (1) Given $(\Gamma_1, T \triangleright) \sim (\Gamma'_1, T' \triangleright)$ and $(\Gamma_1, \Gamma_2 \triangleright) \sim (\Gamma'_1, \Gamma'_2 \triangleright)$ we have to show that

$$(\Gamma_1, T, \partial^n \Gamma_2) \sim (\Gamma'_1, T', \partial^n \Gamma'_2).$$

where $n = l_f(\Gamma_1) = l_f(\Gamma'_1)$.

Proceed by induction on $l_f(\Gamma_2)$. For $l_f(\Gamma_2) = 0$ the assertion is obvious. Let $(\Gamma_1, T \triangleright) \sim (\Gamma'_1, T' \triangleright)$ and $(\Gamma_1, \Gamma_2, S \triangleright) \sim (\Gamma'_1, \Gamma'_2, S' \triangleright)$. The later condition is equivalent to $(\Gamma_1, \Gamma_2 \triangleright) \sim (\Gamma'_1, \Gamma'_2 \triangleright)$ and $(\Gamma_1, \Gamma_2 \triangleright S = S')$. By the inductive assumption we have $(\Gamma_1, T, \partial^n \Gamma_2) \sim (\Gamma'_1, T', \partial^n \Gamma'_2)$. By (5a) we conclude that $(\Gamma_1, T, \partial^n \Gamma_2 \triangleright \partial^n S = \partial^n S')$. Therefore by definition of \sim we have $(\Gamma_1, T, \partial^n \Gamma_2, \partial^n S) \sim (\Gamma'_1, T', \partial^n \Gamma'_2, \partial^n S')$.

(2) Given $(\Gamma_1, T \triangleright) \sim (\Gamma'_1, T' \triangleright)$ and $(\Gamma_1, \Gamma_2 \triangleright o : S) \simeq (\Gamma'_1, \Gamma'_2 \triangleright o' : S')$ we have to show that $(\Gamma_1, T, \partial^n \Gamma_2 \triangleright \partial^n o : \partial^n S) \simeq (\Gamma'_1, T', \partial^n \Gamma'_2 \triangleright \partial^n o' : \partial^n S')$ where $n = l_f(\Gamma_1) = l_f(\Gamma'_1)$. We have $(\Gamma_1, \Gamma_2, S) \sim (\Gamma'_1, \Gamma'_2, S')$ and $(\Gamma_1, \Gamma_2 \triangleright o = o' : S)$. By (5b) we get $(\Gamma_1, T, \partial^n \Gamma_2 \triangleright \partial^n o = \partial^n o' : \partial^n S)$. By (1) of this lemma we get $(\Gamma_1, T, \partial^n \Gamma_2, \partial^n S) \sim (\Gamma'_1, T', \partial^n \Gamma'_2, \partial^n S')$ and therefore by definition of \simeq we get $(\Gamma_1, T, \partial^n \Gamma_2 \triangleright \partial^n o : \partial^n S) \simeq (\Gamma'_1, T', \partial^n \Gamma'_2 \triangleright \partial^n o' : \partial^n S')$.

(3) Given $(\Gamma_1 \triangleright r : T) \simeq (\Gamma'_1 \triangleright r' : T')$ and $(\Gamma_1, T, \Gamma_2 \triangleright) \sim (\Gamma'_1, T', \Gamma'_2 \triangleright)$ we have to show that

$$(\Gamma_1, s_{n+1}(\Gamma_2[r/n + 1])) \sim (\Gamma'_1, s_{n+1}(\Gamma'_2[r'/n + 1])).$$

where $n = l_f(\Gamma_1) = l_f(\Gamma'_1)$. Proceed by induction on $l_f(\Gamma_2)$. For $l_f(\Gamma_2) = 0$ the assertion follows directly from the definitions. Let $(\Gamma_1 \triangleright r : T) \simeq (\Gamma'_1 \triangleright r' : T')$ and $(\Gamma_1, T, \Gamma_2, S \triangleright) \sim (\Gamma'_1, T', \Gamma'_2, S' \triangleright)$.

The later condition is equivalent to $(\Gamma_1, T, \Gamma_2 \triangleright) \sim (\Gamma'_1, T', \Gamma'_2 \triangleright)$ and $(\Gamma_1, T, \Gamma_2 \triangleright S = S')$. By the inductive assumption we have $(\Gamma_1, s_{n+1}(\Gamma_2[r/n+1])) \sim (\Gamma'_1, s_{n+1}(\Gamma'_2[r'/n+1]))$. It remains to show that $(\Gamma_1, s_{n+1}(\Gamma_2[r/n+1]) \triangleright s_{n+1}(S[r/n+1]) = s_{n+1}(S'[r'/n+1]))$. By (2d) it is sufficient to show that $(\Gamma_1, s_{n+1}(\Gamma_2[r/n+1]) \triangleright s_{n+1}(S[r/n+1]) = s_{n+1}(S'[r'/n+1]))$ and $(\Gamma_1, s_{n+1}(\Gamma_2[r/n+1]) \triangleright s_{n+1}(S'[r'/n+1]) = s_{n+1}(S'[r'/n+1]))$. The first relation follows directly from (6a). To prove the second one it is sufficient by (7a) to show that $(\Gamma_1, T, \Gamma_2, S' \triangleright)$ which follows from our assumption through (2c) and (2a).

(4) Given $(\Gamma_1 \triangleright r : T) \simeq (\Gamma'_1 \triangleright r' : T')$ and $(\Gamma_1, T, \Gamma_2 \triangleright o : S) \simeq (\Gamma'_1, T', \Gamma'_2 \triangleright o' : S')$ we have to show that

$$\begin{aligned} & (\Gamma_1, s_{n+1}(\Gamma_2[r/n+1]) \triangleright s_{n+1}(o[r/n+1]) : s_{n+1}(S[r/n+1])) \simeq \\ & (\Gamma'_1, s_{n+1}(\Gamma'_2[r'/n+1]) \triangleright s_{n+1}(o'[r'/n+1]) : s_{n+1}(S'[r'/n+1])). \end{aligned}$$

where $n = l_f(\Gamma_1) = l_f(\Gamma'_1)$ or equivalently that

$$(\Gamma_1, s_{n+1}(\Gamma_2[r/n+1]), s_{n+1}(S[r/n+1])) \sim (\Gamma'_1, s_{n+1}(\Gamma'_2[r'/n+1]), s_{n+1}(S'[r'/n+1]))$$

and $(\Gamma_1, s_{n+1}(\Gamma_2[r/n+1]) \triangleright s_{n+1}(o[r/n+1]) = s_{n+1}(o'[r'/n+1]) : s_{n+1}(S[r/n+1]))$. The first statement follows from part (3) of the lemma. To prove the second statement it is sufficient by (3d) to show that $(\Gamma_1, s_{n+1}(\Gamma_2[r/n+1]) \triangleright s_{n+1}(o[r/n+1]) = s_{n+1}(o'[r'/n+1]) : s_{n+1}(S[r/n+1]))$ and $(\Gamma_1, s_{n+1}(\Gamma_2[r/n+1]) \triangleright s_{n+1}(o'[r'/n+1]) = s_{n+1}(o'[r'/n+1]) : s_{n+1}(S[r/n+1]))$. The first assertion follows directly from (6b). To prove the second one it is sufficient in view of (7b) to show that $(\Gamma_1, T, \Gamma_2 \triangleright o' : S)$ which follows conditions (3c) and (3a).

(5) Given $(\Gamma, T) \sim (\Gamma', T')$ we need to show that $(\Gamma, T \triangleright (n+1) : T) \simeq (\Gamma', T' \triangleright (n+1) : T')$ or equivalently that $(\Gamma, T, T) \sim (\Gamma', T', T')$ and $(\Gamma, T \triangleright (n+1) = (n+1) : T)$. The second part follows from (3b). To prove the first part we need to show that $(\Gamma, T \triangleright T = T')$. This follows from our assumption by (5a).

Lemma 8.7 [2014.07.12.11] *Let C be a subset of $Ob(CC(\mathbf{R}, \mathbf{LM}))$ which is closed under ft. Let \leq be a transitive relation on C such that:*

1. $\Gamma \leq \Gamma'$ implies $l_f(\Gamma) = l_f(\Gamma')$,
2. $\Gamma \in C$ and $ft(\Gamma) \leq F$ implies $\sigma(\Gamma, F) \in C$ and $\Gamma \leq \sigma(\Gamma, F)$.

Then $\Gamma \in C$ and $ft^i(\Gamma) \leq F$ for some $i \geq 1$, implies that $\Gamma \leq \sigma(\Gamma, F)$.

Proof: Simple induction on i .

Lemma 8.8 [2014.07.12.12] *Let C and \leq be as in Lemma 8.7. Then one has:*

1. $(\Gamma, T) \leq (\Gamma, T')$ and $\Gamma \leq \Gamma'$ implies that $(\Gamma, T) \leq (\Gamma', T')$,
2. if \leq is ft-monotone (i.e. $\Gamma \leq \Gamma'$ implies $ft(\Gamma) \leq ft(\Gamma')$) and symmetric then $(\Gamma, T) \leq (\Gamma', T')$ implies that $(\Gamma, T) \leq (\Gamma, T')$.

Proof: The first assertion follows from

$$(\Gamma, T) \leq (\Gamma, T') \leq \sigma((\Gamma, T'), \Gamma') = (\Gamma', T')$$

The second assertion follows from

$$(\Gamma, T) \leq (\Gamma', T') \leq \sigma((\Gamma', T'), \Gamma) = (\Gamma, T')$$

where the second \leq requires $\Gamma' \leq \Gamma$ which follows from ft -monotonicity and symmetry.

Lemma 8.9 [2014.07.12.13] *Let C, \leq be as in Lemma 8.7, let \tilde{B} be a subset of $\tilde{Ob}(CC(\mathbf{R}, \mathbf{LM}))$ and \leq' a transitive relation on \tilde{B} such that:*

1. $\mathcal{J} \leq' \mathcal{J}'$ implies $\partial(\mathcal{J}) \leq \partial(\mathcal{J}')$,
2. $\mathcal{J} \in \tilde{B}$ and $\partial(\mathcal{J}) \leq F$ implies $\tilde{\sigma}(\mathcal{J}, F) \in \tilde{B}$ and $\mathcal{J} \leq' \tilde{\sigma}(\mathcal{J}, F)$.

Then $\mathcal{J} \in \tilde{B}$ and $ft^i(\partial(\mathcal{J})) \leq F$ for some $i \in \mathbf{N}$ implies $\mathcal{J} \leq \tilde{\sigma}(\mathcal{J}, F)$.

Proof: Simple induction on i .

Lemma 8.10 [2014.07.12.14] *Let C, \leq and \tilde{B}, \leq' be as in Lemma 8.9. Then one has:*

1. $(\Gamma \triangleright o : T) \leq' (\Gamma \triangleright o' : T)$ and $(\Gamma, T) \leq (\Gamma', T')$ implies that $(\Gamma \triangleright o : T) \leq' (\Gamma' \triangleright o' : T')$,
2. if (\leq, \leq') is ∂ -monotone (i.e. $\mathcal{J} \leq' \mathcal{J}'$ implies $\partial(\mathcal{J}) \leq \partial(\mathcal{J}')$) and \leq is symmetric then $(\Gamma \triangleright o : T) \leq' (\Gamma' \triangleright o' : T')$ implies that $(\Gamma \triangleright o : T) \leq (\Gamma' \triangleright o' : T)$.

Proof: The first assertion follows from

$$(\Gamma \triangleright o : T) \leq' (\Gamma \triangleright o' : T) \leq' \tilde{\sigma}((\Gamma \triangleright o' : T), (\Gamma', T')) = (\Gamma' \triangleright o' : T')$$

The second assertion follows from

$$\Gamma \triangleright o : T \leq (\Gamma' \triangleright o' : T') \leq' \sigma((\Gamma' \triangleright o' : T'), (\Gamma, T)) = (\Gamma \triangleright o' : T)$$

where the second \leq requires $\Gamma' \leq \Gamma$ which follows from ∂ -monotonicity of \leq' and symmetry of \leq .

Proposition 8.11 [2014.07.10.prop2] *Let (C, \tilde{B}) be subsets in $Ob(CC(\mathbf{R}, \mathbf{LM}))$ and $\tilde{Ob}(CC(\mathbf{R}, \mathbf{LM}))$ respectively which correspond to a C -subsystem CC of $CC(\mathbf{R}, \mathbf{LM})$. Then the constructions presented above establish a bijection between pairs of subsets $(\text{Beq}, \tilde{\text{Beq}})$ which together with (C, \tilde{B}) satisfy the conditions of Proposition 8.2 and pairs of equivalence relations (\sim, \simeq) on (C, \tilde{B}) such that:*

1. (\sim, \simeq) corresponds to a regular congruence relation on CC (i.e., satisfies the conditions of [?, Proposition 5.4]),
2. $\Gamma \in C$ and $ft(\Gamma) \sim F$ implies $\Gamma \sim \sigma(\Gamma, F)$,
3. $\mathcal{J} \in \tilde{B}$ and $\partial(\mathcal{J}) \sim F$ implies $\mathcal{J} \simeq \tilde{\sigma}(\mathcal{J}, F)$.

Proof: One constructs a pair (\sim, \simeq) from $(\text{Req}, \widetilde{\text{Req}})$ as in Definition 8.1. This pair corresponds to a regular congruence relation by Proposition 8.2. Conditions (2),(3) follow from Lemma 8.3.

Let (\sim, \simeq) be equivalence relations satisfying the conditions of the proposition. Define Req as the set of sequences (Γ, T, T') such that $(\Gamma, T), (\Gamma, T') \in C$ and $(\Gamma, T) \sim (\Gamma, T')$. Define $\widetilde{\text{Req}}$ as the set of sequences (Γ, T, o, o') such that $(\Gamma, T, o), (\Gamma, T, o') \in \widetilde{B}$ and $(\Gamma, T, o) \simeq (\Gamma, T, o')$.

Let us show that these subsets satisfy the conditions of Proposition 8.2. Conditions (2.a-2.d) and (3.a-3d) are obvious.

Condition (4a) follows from (2) by Lemma 8.7. Conditions (4b) and (4c) follow from (3) by Lemma 8.9.

Conditions (5a) and (5b) follow from the compatibility of (\sim, \simeq) with T and \widetilde{T} .

Conditions (6a),(6b),(7a),(7b) follow from the compatibility of (\sim, \simeq) with S and \widetilde{S} .