CONSERVATIVITY FOR TYPE THEORIES WITH PROPOSITIONAL COMPUTATION RULES (WIP)

RAFAËL BOCQUET

Contents

Introduction

The goal of these notes is to give proofs of (coherence and) conservativity statements between dependent type theories, more specifically for extensions of type theories with propositional computation rules to type theories with judgmental computation rules. The logical structure of dependent type theories is usually presented by giving introduction and elimination rules, and computation rules that are judgmental, i.e. they completely identify the two terms they equate. The judgmental computation rule for function types is the β rule $(\lambda x \to u)\ v \equiv u[x/v]$. In the presence of identity types, we can use them to obtain propositional computation rules instead of the judgmental ones. The propositional rule for function types would give a family of terms $\lambda_\beta(u,v) : \mathsf{Id}((\lambda x \to u)\ v, u[x/v])$. The type theory with propositional rules can straightforwardly be interpreted in the judgmental one, which is thus stronger. However, we can ask whether the stronger theory is strictly stronger, or whether it is actually conservative over the weaker one.

Such a conservativity statement has been proven for a specific extension of type theories by Martin Hofmann in his thesis, where he considers an extension of intensional type theory to

Date: September 6, 2018.

extensional type theory. A similar result for a variant of the calculus of constructions was proven by Nicolas Oury, using different methods. Both of these proofs construct propositional equalities by induction on the syntax of type theory and rely on the uniqueness of identity proofs (UIP) principle in order to show that any two possible choices of propositional equalities are actually (propositionally) equal. As we are interested in type theories where the UIP principle does not hold, we will need new methods in order to make coherent choices of propositional equalities.

The smallest example of type theory that we will consider has only identity types, and either version of the computation rule for identity types. One of the motivation that bring us to consider this type theory is that the type of paths in cubical type theory satisfies the propositional computation rule, but not the judgmental one.

This work is also relevant in the context of internal language conjectures and the homotopy theory of type theories.

References: The homotopy theory of type theories, Morita equivalences between algebraic dependent type theories, ...
Conservative functors = weak equivalences
Quillen equivalent categories of models

Several notions of categorical models for dependent type theories can be defined as the models of specific generalized algebraic theories, and it is possible to define a general notion of conservativity for generic morphisms of models of generalized algebraic theories. The first result of this document is a characterization of conservativity for equational extensions of generalized algebraic theories. The main part of Hofmann's proof of conservativity can be seen as an instance of this characterization in a specific case.

Relate coherence/strictification and conservativity ?

For type theories with identity types, this characterization will imply that conservativity is equivalent to the existence, for any pair of contexts in the weaker theory that become equal in the stronger theory, of a suitable equivalence between them.

We use this characterization in order to prove that conservativity holds for type theories with identity types and any choice of structure among Σ-type, Π-type and Tarski universe. The main construction of the proof is that of a sufficiently large and coherent family of propositional equalities.

This document is organized as follows. In section 1, we recall some of the background we will need on type theory and its models, and introduce our preferred notations. In section 2, we review the definition of generalized algebraic theory, give some results on quotients of models by congruences, define conservativity for arbitrary morphisms and prove a characterization of it for equational extensions of theories. In section 3, we define generalized algebraic theories corresponding to various type theories, and further characterize conservativity in that setting. We finally prove that conservativity holds for our type theories of interest in section 4.

1. Preliminaries

We will use the following usual notations from type theory:
If $B : A \to \mathbf{Set}$ is a family of sets, we will write ΣB, $\sum_{x:A} B(x)$ or $(x : A) \times B(x)$ for the set of dependent pairs $\{(a, b) \mid a \in A, b \in B(a)\}$, and ΠB, $\prod_{x:A} B(x)$ or $(x : A) \to B(x)$ for the set of dependent functions from A to B.

1.1. Contextual categories.

We will work with contextual categories, used both as a description of the semantics of generalized algebraic theory, and as our notion of categorical models of type theory.

1.2. **Propositional identity types.**

2. Conservativity for generalized algebraic theories

Generalized algebraic theories (GATs) are a generalization of (multisorted) algebraic theories introduced by John Cartmell in his thesis. Algebraic theories can be specified by giving sorts, (total) operations between them, and equations between the equations. While mathematical structures such as groups or rings can be defined as models of some algebraic theories, others, such as categories, can not, due to the presence of dependencies between sorts (morphisms in a category have a source and target), and of partial operations (composition as an operation on pairs of arbitrary morphisms is partial, although it is total if we can define composable pairs of morphisms). GATs are one way to extend algebraic theories in order to also describe such structures, by allowing the specification of dependencies between sorts. In models of the GAT of categories, the sort of morphisms is not interpreted as a set, but rather as a family of sets over pairs of objects. As several notions of models of type theory can be defined as models of specific GATs, their study is relevant to the study of type theory.

Similarly to how the semantics of an algebraic theory is entirely contained in its syntactical category with finite products, the semantics of a generalized algebraic theory is captured by its syntactical contextual category. Although it is often easier to discuss the general semantics of GATs using contextual categories, keeping the syntax in mind helps when dealing with concrete instances of GATs.

Discuss EATs ?

We will define conservativity for arbitrary morphisms of models of a GAT. Informally, if B is a sort of a GAT $\mathbb{T}$ depending on another sort A, and $F : \mathcal{M} \to \mathcal{N}$ is a morphisms of models of $\mathbb{T}$, conservativity is the statement that for any $a : \mathcal{M}_A$ and $b : \mathcal{N}_B(F_A(a))$, there is a $b' : \mathcal{M}_B(a)$ such that $F_B(b') = b$. For instance, in the case of categories, conservativity of a functor $F : \mathcal{C} \to \mathcal{D}$ is the statement that for any two objects $x, y \in \mathcal{C}$, and morphism $f : \mathcal{D}(F(x), F(y))$ in $\mathcal{D}$, there is a morphism $f' : \mathcal{C}(x, y)$ such that $F(f') = f$, i.e. F is a full functor.

2.1. **Definition.** We now define generalized algebraic theories and prove their basic syntactical properties. Our definition is the same as Cartmell's, with a slightly different presentation.

We start by defining untyped presyntax over a signature.

Definition 1 (Signature)**.** *A signature* $(\mathcal{S}, \mathcal{O})$ *is given by a set* $\mathcal{S}$ *of sort symbols, and a set* $\mathcal{O}$ *of operation symbols.*

Definition 2 (Presyntax)**.** *Let* $\Sigma = (\mathcal{S}, \mathcal{O})$ *be a signature. We generate inductively set families* $\mathsf{PreCon}_\Sigma : \mathbf{N} \to \mathbf{Set}$, $\mathsf{PreTy}_\Sigma : \mathbf{N} \to \mathbf{Set}$ *and* $\mathsf{PreTer}_\Sigma : \mathbf{N} \to \mathbf{Set}$ *where the index of* PreCon_Σ *represents the context length, and the indices of* PreTy_Σ *and* PreTer_Σ *represent the number of available variables.*

$$\frac{}{\diamond : \mathsf{PreCon}_\Sigma(0)} \qquad \frac{\Gamma : \mathsf{PreCon}_\Sigma(n) \qquad A : \mathsf{PreTy}_\Sigma(n)}{\Gamma, A : \mathsf{PreCon}_\Sigma(n+1)}$$

$$\frac{n \in \mathbf{N} \qquad s : \mathcal{S} \qquad f : \mathsf{List}(\mathsf{PreTer}_\Sigma(n))}{s(\sigma) : \mathsf{PreTy}_\Sigma(n)} \qquad \frac{n \in \mathbf{N} \qquad i < n}{v_i : \mathsf{PreTer}_\Sigma(n)}$$

$$\frac{n \in \mathbf{N} \qquad o : \mathcal{O} \qquad f : \mathsf{List}(\mathsf{PreTer}_\Sigma(n))}{o(\sigma) : \mathsf{PreTer}_\Sigma(n)}$$

Subsitutions, etc

Definition 3 (GAT-specification). *Let $\Sigma = (\mathcal{S}, \mathcal{O})$ be a signature. A GAT-specification over Σ is the data of:*

- *For each $s \in \mathcal{S}$, an arity $n : \mathbf{N}$ and precontext $\Gamma_s : \mathsf{PreCon}_\Sigma(n)$, representing the rule $\Gamma_s \vdash s\ \mathsf{type}$.*
- *For each $o \in \mathcal{O}$, an arity $n : \mathbf{N}$, a precontext $\Gamma_o : \mathsf{PreCon}_\Sigma(n)$ and a pretype $A_o : \mathsf{PreTy}_\Sigma(n)$, representing the rule $\Gamma_o \vdash o : A_o$.*
- *A set of equation indices $\mathcal{E}$.*
- *For any $e \in \mathcal{E}$, an arity $n : \mathbf{N}$, a precontext $\Gamma_e : \mathsf{PreCon}_\Sigma(n)$, and two preterms $l_e, r_e : \mathsf{PreTer}_\Sigma(n)$, representing the rule $\Gamma_e \vdash l_e \equiv r_e$.*

  Add sort equations

We will not give an explicit name to the components of the rules associated to sort, operation and equation symbols of a GAT-specification, but rather either use the sentence "Let $(\Gamma_s \vdash s\ \mathsf{type})$ be the rule associated to s", or the syntax "$s \in \mathcal{S}(\Gamma_s \vdash s\ \mathsf{type})$".

Definition 4 (Typing relations). *Let $(\mathcal{S}, \mathcal{O}, \mathcal{E}, \mathcal{E}_s)$ be a GAT-specification. We define several inductive families of typing relations:*

$$(\Gamma \vdash) \qquad (\Gamma \vdash A\ \mathsf{type}) \qquad (\Gamma \vdash A \equiv B\ \mathsf{type}) \qquad (\Gamma \vdash a : A) \qquad (\Gamma \vdash a \equiv b : A) \qquad (\Gamma \vdash \delta : \Delta)$$

$$(\Gamma \vdash \delta \equiv \sigma : \Delta)$$

They are given by the following rules:

$$\frac{}{\diamond \vdash} \qquad \frac{\Gamma \vdash \qquad \Gamma \vdash A\ \mathsf{type}}{\Gamma, A \vdash} \qquad \frac{s \in \mathcal{S}(\Gamma_s \vdash s\ \mathsf{type}) \qquad \Gamma_s \vdash \qquad \Gamma \vdash \gamma : \Gamma_s}{\Gamma \vdash s(\gamma)\ \mathsf{type}}$$

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

$$\frac{\Gamma \vdash A \equiv B\ \mathsf{type} \qquad \Delta \vdash \gamma \equiv \sigma : \Gamma}{\Delta \vdash A[\gamma] \equiv B[\sigma]\ \mathsf{type}} \qquad \frac{\Gamma \vdash \qquad \Gamma = (\diamond, A_0, A_1, \ldots, A_{n-1})}{\Gamma \vdash v_i : A_i}$$

$$\frac{o \in \mathcal{O}(\Gamma_o \vdash o : A_o) \qquad \Gamma_o \vdash A_o \qquad \Gamma \vdash \gamma : \Gamma_o}{\Gamma \vdash o(\gamma) \in A_o[\gamma]} \qquad \frac{\Gamma \vdash a : A \qquad \Gamma \vdash A \equiv B\ \mathsf{type}}{\Gamma \vdash a : B}$$

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash a \equiv a : A} \qquad \frac{\Gamma \vdash a \equiv b : A}{\Gamma \vdash b \equiv a : A} \qquad \frac{\Gamma \vdash a \equiv b : A \qquad \Gamma \vdash b \equiv c : A}{\Gamma \vdash a \equiv c : A}$$

$$\frac{e \in \mathcal{E}(\Gamma_e \vdash l_e \equiv r_e) \qquad \Gamma_e \vdash l_e : A \qquad \Gamma_e \vdash r_e : A}{\Gamma_e \vdash l_e \equiv r_e : A}$$

$$\frac{e \in \mathcal{E}_s(\Gamma_e \vdash L_e \equiv R_e) \qquad \Gamma_e \vdash L_e \qquad \Gamma_e \vdash R_e}{\Gamma_e \vdash L_e \equiv R_e} \qquad \frac{\Gamma \vdash a \equiv b : A \qquad \Delta \vdash \gamma \equiv \sigma : \Gamma}{\Delta \vdash a[\gamma] \equiv b[\sigma] : A[\gamma]}$$

$$\frac{\Gamma \vdash}{\Gamma \vdash () : \diamond} \qquad \frac{\Gamma \vdash}{\Gamma \vdash () \equiv () : \diamond} \qquad \frac{\Gamma \vdash \delta : \Delta \qquad \Delta \vdash A\ \mathsf{type} \qquad \Gamma \vdash a : A[\delta]}{\Gamma \vdash (\delta, a) : (\Delta, A)}$$

$$\frac{\Gamma \vdash \delta \equiv \sigma : \Delta \qquad \Delta \vdash A\ \mathsf{type} \qquad \Gamma \vdash a \equiv b : A[\delta]}{\Gamma \vdash (\delta, a) \equiv (\sigma, b) : (\Delta, A)}$$

Definition 5 (Well-formed specification)**.** *Let $(\mathcal{S}, \mathcal{O}, \mathcal{E}, \mathcal{E}_s)$ be a specification. We say that it is well-formed when:*

- *For any $s \in \mathcal{S}(\Gamma_s \vdash s \ \mathsf{type})$, $(\Gamma_s \vdash)$ is derivable.*
- *For any $o \in \mathcal{O}(\Gamma_o \vdash o \in A_o)$, $(\Gamma_o \vdash A_o)$ is derivable.*
- *For any $e \in \mathcal{E}(\Gamma_e \vdash l_e \equiv r_e)$, there exists some pretype A such that $\Gamma_e \vdash l_e : A$ and $\Gamma_e \vdash r_e : A$ are derivable.*
- *For any $e \in \mathcal{E}_s(\Gamma_e \vdash L_e \equiv R_e)$, the judgments $(\Gamma_e \vdash L_e)$ and $(\Gamma_e \vdash R_e)$ are derivable.*

Definition 6 (Generalized algebraic theory)**.** *A generalized algebraic theory is a well-formed specification.*

Whenever we have a GAT $\mathbb{T}$, we can define its syntax as quotients of the presyntax by the convertibility relations.

Definition 7. *Let $\mathbb{T}$ be a GAT. It has a syntactical contextual category $\mathcal{C}_{\mathbb{T}}$ whose objects are contexts of $\mathbb{T}$, and morphisms are substitutions of $\mathbb{T}$.*

State the missing syntactical lemmas... (subsitution, weakening, ...)

2.2. **Semantics.** We now define a notion of model for any contextual category. The models of a GAT are the models of its syntactical contextual category.

Definition 8 (Structure of fl-sketch)**.** *Let $\mathcal{C}$ be any contextual category. We equip it with the structure of a finite limits sketch by picking as distinguished cones the empty context $\diamond$, and all squares of the form:*

$$\begin{array}{ccc} \Delta, f^\star(A) & \xrightarrow{q(f,A)} & \Gamma, A \\ \big\downarrow {\scriptstyle p_{f^\star(A)}} & & \big\downarrow {\scriptstyle p_A} \\ \Delta & \xrightarrow{\quad f \quad} & \Gamma \ . \end{array} \tag{1}$$

Definition 9 (Models of contextual category)**.** *The models of a contextual category $\mathcal{C}$ are the models of $\mathcal{C}$ seen as a fl-sketch. We denote its category of models by $\mathbf{Mod}_{\mathcal{C}}$, or $\mathbf{Mod}_{\mathbb{T}}$ when $\mathcal{C}$ is the syntactical category of a GAT $\mathbb{T}$.*
The objects of $\mathbf{Mod}_{\mathcal{C}}$ are the functors $F : \mathcal{C}_{\mathbb{T}} \to \mathbf{Set}$ that map the distinguished cones of $\mathcal{C}$ to limit cones.
This definition generalizes to categories of models $\mathbf{Mod}_{\mathbb{T}}^{\mathcal{D}}$ in any finitely complete category $\mathcal{D}$.

Mention models in a model category : mapping the squares to homotopy pullbacks, and the display maps to fibrations

Definition 10 (Initial model)**.** *The syntax of a GAT $\mathbb{T}$ gives an explicit description of an initial model. We will call $\mathcal{I}_{\mathbb{T}}$ the initial model of $\mathbb{T}$ defined by:*

$$\mathcal{I}_{\mathbb{T}}(\Gamma) :\equiv \mathcal{C}_{\mathbb{T}}(\diamond, \Gamma),$$

$$\mathcal{I}_{\mathbb{T}}(\sigma : \mathcal{C}_{\mathbb{T}}(\Gamma, \Delta))(f : \mathcal{C}_{\mathbb{T}}(\diamond, \Gamma)) :\equiv (f; \sigma).$$

Proof. Initiality

□

2.3. **Extensions.**

Definition 11 (GAT extension)**.** *We say that a GAT $\mathbb{T}'$ is an extension of a GAT $\mathbb{T}$, and write $\mathbb{T} \subseteq \mathbb{T}'$, when the sort, operation and equation symbols of $\mathbb{T}$ embed in those of $\mathbb{T}'$, and their corresponding rules in $\mathbb{T}$ and $\mathbb{T}'$ match.*
In that case, there is a functor $\iota : \mathcal{C}_{\mathbb{T}} \to \mathcal{C}_{\mathbb{T}'}$ that extends the embeddings to the whole syntax.
By precomposition, ι provides a functor $(- \circ \iota) : \mathbf{Mod}_{\mathbb{T}'} \to \mathbf{Mod}_{\mathbb{T}}$ that allows one to view any model of $\mathbb{T}'$ as a model of $\mathbb{T}$. In particular, the initial model of $\mathbb{T}'$ is a model of $\mathbb{T}$, and the initiality of $\mathcal{I}_{\mathbb{T}}$ provides an unique morphism $F : \mathcal{I}_{\mathbb{T}} \to \mathcal{I}_{\mathbb{T}'}$.
When $\mathbb{T}'$ extends $\mathbb{T}$ by new term equations only, we say that $\mathbb{T}'$ is an equational extension of $\mathbb{T}$, and write $\mathbb{T} \subseteq_{\mathcal{E}} \mathbb{T}'$, where $\mathcal{E}$ is the set of new equation symbols in $\mathbb{T}'$.

Definition 12 (Simple extension)**.** *We say that a GAT extension $\mathbb{T} \subseteq \mathbb{T}'$ is simple when all of the rules corresponding to the new sort, operation and equation symbols are well-formed in $\mathbb{T}$.*

Proposition 1. *Let $\mathbb{T} \subseteq \mathbb{T}'$ be a GAT extension. It can be decomposed as a countable chain of simple extensions $\mathbb{T} = \mathbb{T}_0 \subseteq \mathbb{T}_1 \subseteq \mathbb{T}_2 \subseteq \cdots \subseteq \mathbb{T}_n \subseteq \cdots$, such that $\mathbb{T}'$ is the union of all $(\mathbb{T}_n)_{n\in\mathbf{N}}$. $\mathcal{C}_{\mathbb{T}'}$ is then a colimit of the sequence*

$$\mathcal{C}_{\mathbb{T}_0} \to \mathcal{C}_{\mathbb{T}_1} \to \cdots \to \mathcal{C}_{\mathbb{T}_n} \to \cdots .$$

Proposition 2. *Let $\mathbb{T} \subseteq \mathbb{T}'$ be an equational extension of GATs, and write $\iota : \mathcal{C}_{\mathbb{T}} \to \mathcal{C}_{\mathbb{T}'}$ for the embedding of contextual categories.*
Then $(- \circ \iota) : \mathbf{Mod}_{\mathbb{T}'} \to \mathbf{Mod}_{\mathbb{T}}$ is fully faithful, and a model $\mathcal{M}$ of $\mathbb{T}$ can be extended to a model of $\mathbb{T}'$ if and only if there is a (necessarily unique) morphism in $\mathbf{Mod}_{\mathbb{T}}(\mathcal{I}'_{\mathbb{T}}, \mathcal{M})$.

2.4. **Congruences.** We now focus on the notion of congruences and quotients for GATs. Quotients of GATs always exist, due to the cocompleteness of their categories of models, but they may be more difficult to compute as in the restricted case of algebraic theories. Indeed, whereas a quotient of a model of an algebraic theory by a congruence may simply be computed by taking the quotients for each sort of the theory, this is no longer possible for general GATs, due to the partiality of some operations. This can already be seen for categories, where previously non-composable morphisms may become composable in the quotients, and their composition may not correspond to any morphism from the original model. The specific case of categories has previously been studied by ... in [], and most of their constructions should generalize to all GATs. We will focus here on sufficient conditions that allow a quotient to be computed by taking the quotient for each sort, which will also imply that the quotient inclusion is surjective.

We will denote by $\mathbf{U}$ and $\mathbf{Q}$ the forgetful functor $\mathbf{U} : \mathbf{Setoid} \to \mathbf{Set}$ and the quotient functor $\mathbf{Q} : \mathbf{Setoid} \to \mathbf{Set}$ respectively. For any setoid A, we denote the quotient inclusion by $\mathbf{q}_A : \mathbf{U}(A) \to \mathbf{Q}(A)$.

Definition 13 (Congruence)**.** *Let $\mathcal{C}$ be a contextual category. A congruence on a model $\mathcal{M}$ of $\mathcal{C}$ in* **Set** *is a model $\mathcal{M}'$ of $\mathcal{C}$ in* **Setoid***, such that $(\mathcal{M}'; U) = \mathcal{M}$.*

The categories of models of contextual categories are finitely complete, and we could thus define congruences as internal equivalence relations. We can indeed associate to any congruence an internal equivalence relation.

More details / references

Definition 14 (Associated internal congruence)**.** *Let $\mathcal{C}$ be a contextual category, and $(\mathcal{M}, \sim)$ be a congruence over a model $\mathcal{M}$ of $\mathcal{C}$.*
We define an internal equivalence relation $\mathcal{M} \times_{\sim} \mathcal{M}$ by:

$$(\mathcal{M} \times_{\sim} \mathcal{M})(\Gamma) :\equiv \mathcal{M}(\Gamma) \times_{\sim_\Gamma} \mathcal{M}(\Gamma)$$
$$(\mathcal{M} \times_{\sim} \mathcal{M})(\sigma)(x, y) :\equiv (\mathcal{M}(\sigma)(x), \mathcal{M}(\sigma)(y))$$

Definition 15 (Quotient)**.** *Let $\mathcal{C}$ be a contextual category, and $(\mathcal{M}, \sim)$ be a congruence of $\mathbb{T}$. A quotient of $(\mathcal{M}, \sim)$ is a quotient of the associated internal congruence of $(\mathcal{M}, \sim)$, i.e. a coequalizer of*

$$\mathcal{M} \times_{\sim} \mathcal{M} \underset{\pi_2}{\overset{\pi_1}{\rightrightarrows}} \mathcal{M} \ . \tag{2}$$

Although quotients always exist, they may not in general be computed pointwise, and the quotient inclusion morphism does not have to be surjective. However, when the congruence sends display maps to setoid fibrations, both of these properties hold.

Definition 16 (Setoid fibration)**.** *Let $f : (B, \sim_B) \to (A, \sim_A)$ be a setoid morphism. If $a \in A$, we denote by $B(a)$ the set of points of B whose image by f is a.*
We say that f is a setoid fibration when one of the following equivalent properties hold:

- *f has the right lifting property with respect to $\{x\} \to \{x \sim y\}$.*
- *$\forall b \in B, a \in A, a \sim_A f(b) \implies \exists b' \in B, f(b') = a \wedge b \sim b'$.*
- *f is equipped with a family of transport operations $p_f : \prod_{x \sim_A y} (B(x) \to B(y))$ such that $\forall x \sim_B y, b \in B(x), p_{f,x,y}(b) \sim_B b$.*

Lemma 1 (Pullbacks along setoid fibrations)**.** *Any pullback in* **Setoid** *of which at least one leg is a fibration is preserved by the quotient functor* $\mathbf{Q}$.

Proof. Let

$$\begin{array}{ccc} (A \times_C B, \sim) & \xrightarrow{\pi_A} & (A, \sim_A) \\ \downarrow{\scriptstyle \pi_B} & \lrcorner & \downarrow{\scriptstyle f} \\ (B, \sim_B) & \xrightarrow{g} & (C, \sim_C) \end{array} \tag{3}$$

be a pullback in **Setoid**, and assume that f is a fibration. The relation $\sim$ on $A \times_C B$ is defined by $(a_1, b_1) \sim (a_2, b_2) \iff a_1 \sim a_2 \wedge b_1 \sim b_2$.
We have in general an injective map $r : \mathbf{Q}(A \times_C B) \to \mathbf{Q}\ A \times_{\mathbf{Q}\ C} \mathbf{Q}\ B$ defined by $r(\mathbf{q}(a, b)) :\equiv (\mathbf{q}\ a, \mathbf{q}\ b)$. Let $p : \prod_{x \sim_C y} (A(x) \to A(y))$ be a family of transport operations for f. We can define a map $r^{-1} : \mathbf{Q}\ A \times_{\mathbf{Q}\ C} \mathbf{Q}\ B \to \mathbf{Q}(A \times_C B)$ by $r^{-1}(\mathbf{q}\ a, \mathbf{q}\ b) :\equiv \mathbf{q}(p_{f(a),g(b)}(a), b)$. One can check that r^{-1} is well-defined and that r and r^{-1} are inverses. □

Theorem 1. *Let $\mathcal{M} : \mathcal{C} \to$ **Setoid** be a congruence of a contextual category $\mathcal{C}$. If every display map of $\mathcal{C}$ is mapped by $\mathcal{M}$ to a setoid fibration, then the functor $(\mathcal{M}; \mathbf{Q}) : \mathcal{C} \to$ **Set** is a model of $\mathcal{C}$ and a quotient of $\mathcal{M}$. In that case, we also have that the quotient inclusion $\mathbf{q}_{\mathcal{M}} : (\mathcal{M}; \mathbf{U}) \to (\mathcal{M}; \mathbf{Q})$ is surjective.*

Proof. To prove that $(\mathcal{M}; \mathbf{Q})$ is a model of $\mathcal{C}$, we have to show that it sends the distinguished cones of $\mathcal{C}$ to limit cones. This is trivial for the empty context $\diamond$, since $\mathcal{M}(\diamond)$ is terminal. $\mathcal{M}$ sends the distinguished squares of $\mathcal{C}$ to squares whose right map is of the form $\mathcal{M}(p)$ for some display map p. By hypothesis, those maps are setoid fibrations, and $\mathbf{Q}$ maps these squares to pullback squares by lemma 1. Thus $(\mathcal{M}; \mathbf{Q})$ is indeed a model of $\mathcal{C}$.
Then since $(\mathcal{M}; \mathbf{U}) \xrightarrow{\mathbf{q}_{\mathcal{M}}} (\mathcal{M}; \mathbf{Q})$ is a quotient in the category of presheaves $\widehat{\mathcal{C}^{op}}$ and $\mathbf{Mod}_{\mathcal{C}}$ is a reflexive subcategory of $\widehat{\mathcal{C}^{op}}$, $\mathbf{q}_{\mathcal{M}}$ is also a quotient in $\mathbf{Mod}_{\mathcal{C}}$. □

If we have a presentation of the contextual category by a GAT, then it is enough to ask that all sort projection maps are mapped to setoid fibrations, since fibrations are closed by composition and arbitrary pullbacks.

2.5. Conservativity.

Definition 17. *Let $\mathcal{C}$ be a contextual category, and $F : \mathcal{M} \to \mathcal{N}$ a morphism of $\mathbf{Mod}_{\mathcal{C}}$.*
We say that F is conservative at a display map $p : \Gamma.\Delta \to \Gamma$ if for any $\gamma \in \mathcal{M}(\Gamma)$ and $\delta \in \mathcal{N}(\Gamma.\Delta)$ such that $\mathcal{N}(p)(\delta) = F_\Gamma(\gamma)$, there is a $\delta' \in \mathcal{M}(\Gamma.\Delta)$ such that $F_{\Gamma.\Delta}(\delta') = \delta$.
We say that F is conservative when it is conservative at all display maps.

The following proposition provides an equivalent definition of conservativity. That alternative definition has the advantage of also being valid in the context of models in finitely complete categories other than **Set**.

Proposition 3. *A morphism $F : \mathcal{M} \to \mathcal{N}$ of models of a contextual category $\mathcal{C}$ is conservative at a display map $p : \Gamma.\Delta \to \Gamma$ if and only if the naturality square*

$$\begin{array}{ccc} \mathcal{M}(\Gamma.\Delta) & \xrightarrow{F(\Gamma.\Delta)} & \mathcal{N}(\Gamma.\Delta) \\ \downarrow{\scriptstyle \mathcal{M}(p)} & & \downarrow{\scriptstyle \mathcal{N}(p)} \\ \mathcal{M}(\Gamma) & \xrightarrow{F(\Gamma)} & \mathcal{N}(\Gamma) \end{array} \tag{4}$$

is a pullback square.

Proof. TODO

□

The following proposition implies that to check that a morphism of models of a GAT is conservative, it is enough to look at the sort projection maps.

Proposition 4. *The class of display maps that a model morphism is conservative at is closed by composition and pullbacks along arbitrary morphisms.*

Proof. TODO

□

Definition 18. *Let $F : \mathcal{M} \to \mathcal{N}$ be a morphism of models of a contextual category $\mathcal{C}$.*
The kernel pair of F is an internal congruence on $\mathcal{M}$. We will denote by $(\sim^F) : \prod_{\Gamma \in \mathcal{C}} \mathsf{EqRel}(\mathcal{M}(\Gamma))$ the corresponding family of equivalence relations. They are given by

$$\gamma \sim^F_\Gamma \gamma' \iff F_\Gamma(\gamma) = F_\Gamma(\gamma').$$

Proposition 5. *For any conservative morphism $F : \mathcal{M} \to \mathcal{N}$ of models of a given contextual category $\mathcal{C}$, the congruence $(\mathcal{M}, \sim^F) : \mathcal{C} \to$ **Setoid** maps the display maps of $\mathcal{C}$ to setoid fibrations, i.e. it satisfies the premises of theorem 1.*

Proof. TODO

□

When the model morphism comes from an equational extension of GATs, the converse implication also holds. The proof of this will need the following lemma.

Lemma 2. *Let $\mathbb{T} \subseteq \mathbb{T}'$ be a simple equational extension of GATs, and let $\mathcal{M}$ be a model of $\mathbb{T}'$. Consider the following diagram*

(5)
$$
\begin{array}{ccccc}
 & & Q & & \\
 & & \uparrow{\scriptstyle q} & & \\
 & & \mathcal{I}_{\mathbb{T}} & \xrightarrow{F_0} \mathcal{I}_{\mathbb{T}'} & \xrightarrow{F} \mathcal{M} \\
 & & {\scriptstyle \pi_1}\upuparrows{\scriptstyle \pi_2} & {\scriptstyle \pi_1}\upuparrows{\scriptstyle \pi_2} & \\
 & & \mathsf{kp}(F_0; F) & \longrightarrow \mathsf{kp}(F) & ,
\end{array}
$$

where F_0 and F are obtained from the initiality of $\mathcal{I}_{\mathbb{T}}$ and $\mathcal{I}_{\mathbb{T}'}$, and $q : \mathcal{I}_{\mathbb{T}} \to Q$ is any surjective map such that $\pi_1; q = \pi_2; q$.
Then there is a model Q' of $\mathbb{T}'$, extending Q, such that the unique morphism $q' : \mathbf{Mod}_{\mathbb{T}'}(\mathcal{I}_{\mathbb{T}'}, Q')$ is surjective, and satisfies $\pi_1; q' = \pi_2; q'$.

Theorem 2. *Let $\mathbb{T}'$ be an equational extension of a GAT $\mathbb{T}$, and denote by F the unique morphism $F : \mathcal{I}_{\mathbb{T}} \to \mathcal{I}_{\mathbb{T}'}$.*
*If $(\mathcal{I}_{\mathbb{T}}, \sim^F) : \mathcal{C}_{\mathbb{T}} \to$ **Setoid** maps all sort projection maps to setoid fibrations, then F is conservative.*

Proof. Our assumption is exactly the condition of theorem 1, which implies that the congruence $(\mathcal{I}_{\mathbb{T}}, \sim^F)$ has a quotient $\widetilde{\mathcal{I}_{\mathbb{T}}} : \mathbf{Mod}_{\mathbb{T}}$, and that the quotient inclusion $q : \mathcal{I}_{\mathbb{T}} \to \widetilde{\mathcal{I}_{\mathbb{T}}}$ is a surjection. We will prove that this quotient is isomorphic to $\mathcal{I}_{\mathbb{T}'}$.
Our first step is to prove that it is a model of $\mathbb{T}'$, in the sense that there is a morphism $\iota : \mathcal{I}'_{\mathbb{T}} \to \widetilde{\mathcal{I}_{\mathbb{T}}}$ in $\mathbf{Mod}_{\mathbb{T}}$. The extension $\mathbb{T} \subseteq \mathbb{T}'$ can be decomposed into a sequence of simple equational extensions $\mathbb{T} = \mathbb{T}_0 \subseteq \mathbb{T}_1 \subseteq \cdots \mathbb{T}_n \cdots$. We prove by induction that for any $n \in \mathbf{N}$, there is a morphism $\iota_n : \mathcal{I}_{\mathbb{T}_n} \to \widetilde{\mathcal{I}_{\mathbb{T}}}$. We can set $\iota_0 = \mathbf{q}$, and the inductive step is a consequence of lemma 2. Because $\mathcal{I}_{\mathbb{T}'} = \operatorname{colim} \mathcal{I}_{\mathbb{T}_n}$, this proves that there is a morphism $\iota : \mathcal{I}_{\mathbb{T}'} \to \widetilde{\mathcal{I}_{\mathbb{T}}}$.
The current situation is as follows:

(6)
$$
\begin{array}{ccccc}
 & & & \widetilde{\mathcal{I}_{\mathbb{T}}} & \\
 & & \nearrow{\scriptstyle \mathbf{q}} & & {\scriptstyle \iota}\nwarrow\searrow{\scriptstyle \widetilde{F}} \\
\mathcal{I}_{\mathbb{T}} \times_{\sim} \mathcal{I}_{\mathbb{T}} & \overset{\pi_1}{\underset{\pi_2}{\rightrightarrows}} & \mathcal{I}_{\mathbb{T}} & \xrightarrow{\quad F \quad} & \mathcal{I}_{\mathbb{T}'} ,
\end{array}
$$

where $\widetilde{\mathcal{I}_{\mathbb{T}}} \xrightarrow{\widetilde{F}} \mathcal{I}_{\mathbb{T}'}$ is obtained from the initiality of $\widetilde{\mathcal{I}_{\mathbb{T}}}$ as a quotient.
Because $\mathbf{Mod}_{\mathbb{T}'}$ is a full subcategory of $\mathbf{Mod}_{\mathbb{T}}$, both $\widetilde{F}$ and ι are morphisms of $\mathbf{Mod}_{\mathbb{T}'}$. Since $\mathcal{I}_{\mathbb{T}'}$ is initial, we have $\iota; \widetilde{F} = \mathsf{id}$.
Since $\mathcal{I}_{\mathbb{T}}$ is initial, we have that $F; \iota = \mathbf{q}$. Furthermore, we have from the definition of $\widetilde{F}$ that $F = \mathbf{q}; \widetilde{F}$. As $\mathbf{q}$ is an epimorphism, we therefore have that $\widetilde{F}; \iota = \mathsf{id}$.
This proves that $\widetilde{\mathcal{I}_{\mathbb{T}}}$ and $\mathcal{I}_{\mathbb{T}'}$ are isomorphic, and hence that F is conservative if and only if $\mathbf{q}$ is conservative.
We finish by proving that $\mathbf{q}$ is conservative. Let $s \in \mathcal{S}(\Gamma_s \vdash s \ \mathsf{type})$ be a sort of $\mathbb{T}$, and take $\gamma \in \mathcal{I}_{\mathbb{T}}(\Gamma_s)$ and $a \in \widetilde{\mathcal{I}_{\mathbb{T}}}(s)(\mathbf{q}_\Gamma(\gamma))$. From the surjectivity of $\mathbf{q}$, we obtain $\gamma' : \mathcal{I}_{\mathbb{T}}(\Gamma_s)$ and $a' : \mathcal{I}_{\mathbb{T}}(s)(\gamma')$, such that $\mathbf{q}_{\Gamma_s}(\gamma') = \mathbf{q}_{\Gamma_s}(\gamma)$ and $\mathbf{q}_s(\gamma', a') = a$. Since $\mathbf{q}_{\Gamma_s}(\gamma') = \mathbf{q}_{\Gamma_s}(\gamma)$ implies that $F(\gamma) = F(\gamma')$, we can use the fact that $\mathcal{I}_{\mathbb{T}}(p_s) : \mathcal{I}_{\mathbb{T}}(\Gamma_s, s) \to \mathcal{I}_{\mathbb{T}}(\Gamma_s)$ is a setoid fibration in order to transport a' to $a'' : \mathcal{I}_{\mathbb{T}}(\Gamma_s)(\gamma)$, and check that $\mathbf{q}(a'') = a$. □

Example: Categories, Monoidal categories

3. Type theories as generalized algebraic theories

In this section, we introduce GATs describing models of type theory, with the aim of applying theorem 2 to prove conservativity results between various type theories. In particular, we define a GAT $\mathbb{T}_{\mathsf{CxI}}$ whose category of models is equivalent to the category of contextual categories, and extensions of this GAT corresponding to logical structures commonly found in type theory.
This presentation is very close to Isaev's notion of algebraic dependent type theory, but using GATs rather than partial Horn theories. We believe that most notions and constructions on algebraic dependent type theories should carry over to our setting.

Discuss type theories without definitional computation rules ?

In that setting, we give a further characterization of conservativity for extensions of $\mathbb{T}_{\mathsf{CxI}}$, using the fact that the transport operations needed to prove that functions are setoid fibrations can naturally be defined by substitution in a type-theoretic setting.
Finally, as a first application of this characterization, we use it to give another proof of the conservativity of extensional type theory over intensional type theory.

3.1. The GAT $\mathbb{T}_{\mathsf{CxI}}$.

3.2. Conservativity for type theories.

3.3. Conservativity of TTE over TTI.

4. Conservativity for type theories with propositional computation rules

Check if functional extensionality is needed

In this section, we state and prove our main results: for type theory with identity types and any selection of logical structure among Σ-types, Π-types and a hierarchy of Tarski universes, the type theory with definitional computation rules is conservative over the propositional variant.
The main difficulty in this proof lies in constructing a coherent family of equivalences and (dependent) equalities to circumvent the lack of the UIP principle.

Through this whole section, we assume fixed a choice $\mathcal{L} \subseteq \{\Sigma, \Pi, \mathbf{U}\}$ of logical structure to include in our type theories.

Find a nice way to explain which parts of the proof depend on Σ, Π and $\mathbf{U}$

4.1. Definitions.

Add Sigma,Pi,U

Definition 19 (Type theory with propositional computation rule)**.** *(define the type theory* $\mathbb{T}^w$ *here) (Base type* X*).*

Definition 20 (Type theory with definitional computation rules)**.** *(define the type theory* $\mathbb{T}^s$ *here)*

Proposition 6 (Stripping map)**.** $|-| : \mathbb{T}^w \to \mathbb{T}^s$.

Definition 21. *We generate equivalence relations* $(\sim_{\mathsf{Con}}) : \mathsf{EqRel}(\mathsf{Con}^w)$, $(\sim_{\mathsf{Ty}}) : \mathsf{EqRel}(\Sigma\mathsf{Ty}^w)$ *and* $(\sim_{\mathsf{Ter}}) : \mathsf{EqRel}(\Sigma\mathsf{Ter}^w)$ *by the following rules (in addition to reflexivity, symmetry and transitivity closure):*

Id-CONGR

$$\frac{\Gamma_1 \sim \Gamma_2 \qquad (\Gamma_1 \overset{w}{\vdash} A_1) \sim (\Gamma_2 \overset{w}{\vdash} A_2) \qquad (\Gamma_1 \overset{w}{\vdash} x_1 : A_1) \sim (\Gamma_1 \overset{w}{\vdash} x_2 : A_2) \qquad (\Gamma_1 \overset{w}{\vdash} y_1 : A_1) \sim (\Gamma_1 \overset{w}{\vdash} y_2 : A_2)}{(\Gamma_1 \overset{w}{\vdash} \mathsf{Id}(A_1, x_1, y_1)) \sim (\Gamma_2 \overset{w}{\vdash} \mathsf{Id}(A_2, x_2, y_2))}$$

refl-CONGR

$$\frac{\Gamma_1 \sim \Gamma_2 \qquad (\Gamma_1 \overset{w}{\vdash} A_1) \sim (\Gamma_2 \overset{w}{\vdash} A_2) \qquad (\Gamma_1 \overset{w}{\vdash} x_1 : A_1) \sim (\Gamma_1 \overset{w}{\vdash} x_2 : A_2)}{(\Gamma_1 \overset{w}{\vdash} \mathsf{refl}(x_1) : \mathsf{Id}(A_1, x_1, x_1)) \sim (\Gamma_2 \overset{w}{\vdash} \mathsf{refl}(x_2) : \mathsf{Id}(A_2, x_2, x_2))}$$

J-CONGR

$$\frac{\begin{array}{c}\Gamma_1 \sim \Gamma_2 \qquad (\Gamma_1 \overset{w}{\vdash} A_1) \sim (\Gamma_2 \overset{w}{\vdash} A_2) \\ (\Gamma_1 \overset{w}{\vdash} x_1 : A_1) \sim (\Gamma_1 \overset{w}{\vdash} x_2 : A_2) \qquad (\Gamma_1, (y : A_1), (p : \mathsf{Id}(x_1, y)) \overset{w}{\vdash} \Delta_1) \sim (\Gamma_2, (y : A_2), (p : \mathsf{Id}(x_2, y)) \overset{w}{\vdash} \Delta_2) \\ (\Gamma_1, (y : A_1), (p : \mathsf{Id}(x_1, y)), (\delta : \Delta_1) \overset{w}{\vdash} C_1) \sim (\Gamma_2, (y : A_2), (p : \mathsf{Id}(x_2, y)), (\delta : \Delta_2) \overset{w}{\vdash} C_2) \qquad (\Gamma_1 \overset{w}{\vdash} y_1 : A_1) \sim (\Gamma_2 \overset{w}{\vdash} y_2 : A_2) \\ (\Gamma_1 \overset{w}{\vdash} p_1 : \mathsf{Id}(x_1, y_1)) \sim (\Gamma_2 \overset{w}{\vdash} p_2 : \mathsf{Id}(x_2, y_2)) \qquad (\Gamma_1 \overset{w}{\vdash} \delta_1 : \Delta(y_1, p_1)) \sim (\Gamma_2 \overset{w}{\vdash} \delta_2 : \Delta(y_2, p_2))\end{array}}{(\Gamma_1 \overset{w}{\vdash} \mathsf{J}^{\Delta_1, C_1}_{A_1, x_1}(d_1, y_1, p_1, \delta_1) : C_1(y_1, p_1, \delta_1)) \sim (\Gamma_2 \overset{w}{\vdash} \mathsf{J}^{\Delta_2, C_2}_{A_2, x_2}(d_2, y_2, p_2, \delta_2) : C_2(y_2, p_2, \delta_2))}$$

J_β-CONGR

$$\frac{\begin{array}{c}\Gamma_1 \sim \Gamma_2 \qquad (\Gamma_1 \overset{w}{\vdash} A_1) \sim (\Gamma_2 \overset{w}{\vdash} A_2) \\ (\Gamma_1 \overset{w}{\vdash} x_1 : A_1) \sim (\Gamma_1 \overset{w}{\vdash} x_2 : A_2) \qquad (\Gamma_1, (y : A_1), (p : \mathsf{Id}(x_1, y)) \overset{w}{\vdash} \Delta_1) \sim (\Gamma_2, (y : A_2), (p : \mathsf{Id}(x_2, y)) \overset{w}{\vdash} \Delta_2) \\ (\Gamma_1, (y : A_1), (p : \mathsf{Id}(x_1, y)), (\delta : \Delta_1) \overset{w}{\vdash} C_1) \sim (\Gamma_2, (y : A_2), (p : \mathsf{Id}(x_2, y)), (\delta : \Delta_2) \overset{w}{\vdash} C_2) \\ (\Gamma_1 \overset{w}{\vdash} \delta_1 : \Delta(x_1, \mathsf{refl}(x_1))) \sim (\Gamma_2 \overset{w}{\vdash} \delta_2 : \Delta(x_2, \mathsf{refl}(x_2)))\end{array}}{(\Gamma_1 \overset{w}{\vdash} \mathsf{J}^{\Delta_1, C_1}_{\beta, A_1, x_1}(d_1, \delta_1) : \mathsf{Id}(\mathsf{J}^{\Delta_1, C_1}_{A_1, x_1}(d_1, x_1, \mathsf{refl}(x_1), \delta_1), d_1(\delta_1))) \sim (\Gamma_2 \overset{w}{\vdash} \mathsf{J}^{\Delta_2, C_2}_{\beta, A_2, x_2}(d_2, \delta_2) : \mathsf{Id}(\mathsf{J}^{\Delta_2, C_2}_{A_2, x_2}(d_2, x_2, \mathsf{refl}(x_2), \delta_2), d_2(\delta_2)))}$$

J-EQ

$$\frac{}{(\Gamma \overset{w}{\vdash} \mathsf{J}^{\Delta, C}_{A, x}(d, x, \mathsf{refl}(x), \delta) : C(x, \mathsf{refl}(x), \delta)) \sim (\Gamma \overset{w}{\vdash} d(\delta) : C(x, \mathsf{refl}(x), \delta))}$$

J_β-EQ

$$\frac{}{(\Gamma \overset{w}{\vdash} \mathsf{J}^{\Delta, C}_{\beta, A, x}(d, \delta) : \mathsf{Id}(\mathsf{J}^{\Delta, C}_{A, x}(d, x, \mathsf{refl}(x), \delta), d(\delta))) \sim (\Gamma \overset{w}{\vdash} \mathsf{refl}(d(\delta)) : \mathsf{Id}(d(\delta), d(\delta)))}$$

Proposition 7. *Let* x, y *be two contexts, types or terms of* $\mathbb{T}^w$. *Then* $x \sim y$ *if and only if* $|x| = |y|$.

Proposition 8. $\mathbb{T}^s$ *is conservative over* $\mathbb{T}^w$ *if and only if for any pair* $\Gamma \sim \Delta$ *of congruent contexts of* $\mathbb{T}^w$, *there is an equivalence* $w : \Gamma \simeq \Delta$, *such that* $|w| = \mathsf{id}$.

4.2. The type theory $\mathbb{T}^{\mathsf{coh}}$.

Motivation, relation to Brunerie's weak ∞-groupoids

We now introduce a type theory $\mathbb{T}^{\mathsf{coh}}$, interpretable in $\mathbb{T}^w$, that will be used to construct equalities between congruent terms of $\mathbb{T}^w$ in a coherent way. $\mathbb{T}^{\mathsf{coh}}$ is built from $\mathbb{T}^w$ by freely adding weak identity types, in a specific way. Because $\mathbb{T}^w$ is already equipped with identity types, we will denote the constructors for the new identity types of $\mathbb{T}^{\mathsf{coh}}$ by Coh, $\overset{\mathsf{coh}}{\mathsf{refl}}$, $\overset{\mathsf{coh}}{\mathsf{J}}$ and $\overset{\mathsf{coh}}{\mathsf{J}}_\beta$, and call these new types coherence types.

(Coherence type,coherence) ⇒ (Coherent identity type, Coherent equality/identity/path) ???

We will prove that $\mathbb{T}^{\mathsf{coh}}$ satisfies an external variant of the UIP principle: for any context Γ coming from $\mathbb{T}^w$, any basic type A in Γ, any term x of type A, and any term p of the coherence type $\mathsf{Coh}(A, x, x)$, there is a term of the coherence type $\mathsf{Coh}(\mathsf{Coh}(A, x, x), p, \overset{\mathsf{coh}}{\mathsf{refl}}(x))$.

$\mathbb{T}^{\mathsf{coh}}$ is built from $\mathbb{T}^w$ in the following way. First, we mark all of the type constructors coming from $\mathbb{T}^w$ $(\mathsf{X}, \mathsf{Id}, \ldots)$ as *basic* type constructors, and require all types occurring in the premises of the type and terms constructors of $\mathbb{T}^w$ to be basic. At this point, we have a type theory isomorphic to $\mathbb{T}^w$, with no way to construct non-basic types. We then add coherence types to this theory, with the usual rules of weak identity types. The types of the theory are now the basic types and their iterated congruence types. We also add Σ-types and Π-types, whose main purpose it to simplifying the proof by moving some of the complexity away from the main part of the proof (Interpreting $\mathbb{T}^{\mathsf{coh}}$ into $\mathbb{T}^w$ becomes more complicated instead). Finally, we add terms constructor of the coherence types, correponding to each of the propositional computation rules of $\mathbb{T}^w$.

Definition 22. *We define a type theory $\mathbb{T}^{\mathsf{coh}}$, generated by the following rules:*

$$\frac{\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{type}}{\Gamma, A \ \mathsf{ctx}} \qquad \frac{}{\diamond \ \mathsf{ctx}} \qquad \frac{\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{basic\ type}}{\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{type}}$$

X-INTRO
$$\frac{\Gamma \ \mathsf{ctx}}{\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{X} \ \mathsf{basic\ type}}$$

Id-INTRO
$$\frac{\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} x : A \qquad \Gamma \overset{\mathsf{coh}}{\vdash} y : A}{\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{Id}(A, x, y) \ \mathsf{basic\ type}}$$

Coh-INTRO
$$\frac{\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} x : A \qquad \Gamma \overset{\mathsf{coh}}{\vdash} y : A}{\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{Coh}(A, x, y) \ \mathsf{type}}$$

VAR
$$\frac{\Gamma \ \mathsf{ctx}}{\Gamma \overset{\mathsf{coh}}{\vdash} x_i : A_i}$$

refl-INTRO
$$\frac{\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} x : A}{\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{refl}(x) : \mathsf{Id}(A, x, x)}$$

J-INTRO
$$\frac{\begin{array}{c}\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} x : A \qquad \Gamma, (y : A), (p : \mathsf{Id}(x, y)) \overset{\mathsf{coh}}{\vdash} \Delta \ \mathsf{basic\ ctx} \\ \Gamma, (y : A), (p : \mathsf{Id}(x, y)), (\delta : \Delta(y, p)) \overset{\mathsf{coh}}{\vdash} C \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} y : A \qquad \Gamma \overset{\mathsf{coh}}{\vdash} p : \mathsf{Id}(x, y) \qquad \Gamma \overset{\mathsf{coh}}{\vdash} \delta : \Delta(y, p)\end{array}}{\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{J}^{\Delta, C}_{A, x}(x, y, p, \delta) : C(y, p, \delta)}$$

J_β-INTRO
$$\frac{\begin{array}{c}\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} x : A \\ \Gamma, (y : A), (p : \mathsf{Id}(x, y)) \overset{\mathsf{coh}}{\vdash} \Delta \ \mathsf{basic\ ctx} \qquad \Gamma, (y : A), (p : \mathsf{Id}(x, y)), (\delta : \Delta(y, p)) \overset{\mathsf{coh}}{\vdash} C \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} \delta : \Delta(x, \mathsf{refl}(x))\end{array}}{\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{J}^{\Delta, C}_{\beta, A, x}(x, \delta) : \mathsf{Id}(\mathsf{J}^{\Delta, C}_{A, x}(x, x, \mathsf{refl}(x), \delta), d(\delta))}$$

$\overset{\mathsf{coh}}{\mathsf{refl}}$-INTRO
$$\frac{\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} x : A}{\Gamma \overset{\mathsf{coh}}{\vdash} \overset{\mathsf{coh}}{\mathsf{refl}}(x) : \mathsf{Coh}(A, x, x)}$$

$\overset{\mathsf{coh}}{\mathsf{J}}$-INTRO
$$\frac{\begin{array}{c}\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} x : A \qquad \Gamma, (y : A), (p : \mathsf{Coh}(x, y)) \overset{\mathsf{coh}}{\vdash} \Delta \ \mathsf{ctx} \\ \Gamma, (y : A), (p : \mathsf{Coh}(x, y)), (\delta : \Delta(y, p)) \overset{\mathsf{coh}}{\vdash} C \ \mathsf{type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} y : A \qquad \Gamma \overset{\mathsf{coh}}{\vdash} p : \mathsf{Coh}(x, y) \qquad \Gamma \overset{\mathsf{coh}}{\vdash} \delta : \Delta(y, p)\end{array}}{\Gamma \overset{\mathsf{coh}}{\vdash} \overset{\mathsf{coh}}{\mathsf{J}}{}^{\Delta, C}_{A, x}(x, y, p, \delta) : C(y, p, \delta)}$$

$\overset{\mathsf{coh}}{\mathsf{J}}_\beta$-INTRO
$$\frac{\begin{array}{c}\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{type} \\ \Gamma \overset{\mathsf{coh}}{\vdash} x : A \qquad \Gamma, (y : A), (p : \mathsf{Coh}(x, y)) \overset{\mathsf{coh}}{\vdash} \Delta \ \mathsf{ctx} \qquad \Gamma, (y : A), (p : \mathsf{Coh}(x, y)), (\delta : \Delta(y, p)) \overset{\mathsf{coh}}{\vdash} C \ \mathsf{type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} \delta : \Delta(x, \mathsf{refl}(x))\end{array}}{\Gamma \overset{\mathsf{coh}}{\vdash} \overset{\mathsf{coh}}{\mathsf{J}}{}^{\Delta, C}_{\beta, A, x}(x, \delta) : \mathsf{Coh}(\mathsf{J}^{\Delta, C}_{A, x}(x, x, \mathsf{refl}(x), \delta), d(\delta))}$$

e_{J}-INTRO
$$\frac{\begin{array}{c}\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} x : A \\ \Gamma, (y : A), (p : \mathsf{Id}(x, y)) \overset{\mathsf{coh}}{\vdash} \Delta \ \mathsf{basic\ ctx} \qquad \Gamma, (y : A), (p : \mathsf{Id}(x, y)), (\delta : \Delta(y, p)) \overset{\mathsf{coh}}{\vdash} C \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} \delta : \Delta(x, \mathsf{refl}(x))\end{array}}{\Gamma \overset{\mathsf{coh}}{\vdash} e^{\Delta, C}_{\mathsf{J}, A, x}(d, \delta) : \mathsf{Coh}(\mathsf{J}^{\Delta, C}_{A, x}(x, x, \mathsf{refl}(x), \delta), \underset{\cdot}{(}\delta))}$$

e_{J_β}-INTRO
$$\frac{\begin{array}{c}\Gamma \overset{\mathsf{coh}}{\vdash} A \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} x : A \\ \Gamma, (y : A), (p : \mathsf{Id}(x, y)) \overset{\mathsf{coh}}{\vdash} \Delta \ \mathsf{basic\ ctx} \qquad \Gamma, (y : A), (p : \mathsf{Id}(x, y)), (\delta : \Delta(y, p)) \overset{\mathsf{coh}}{\vdash} C \ \mathsf{basic\ type} \qquad \Gamma \overset{\mathsf{coh}}{\vdash} \delta : \Delta(x, \mathsf{refl}(x))\end{array}}{\Gamma \overset{\mathsf{coh}}{\vdash} e^{\Delta, C}_{\mathsf{J}_\beta, A, x}(d, \delta) : \mathsf{Coh}(\mathsf{J}^{\Delta, C}_{\beta, A, x}(x, \delta), \overset{\mathsf{coh}}{\mathsf{J}}{}^{-}_{-,-}(\mathsf{refl}(d(\delta)), -, e^{\Delta, C}_{\mathsf{J}, A, x}(d, \delta)))}$$

$$\frac{}{XXXXXXXXX}\ (\text{Rules for } \mathbf{1}) \qquad \frac{}{XXXXXXXXX}\ (\text{Rules for } \Sigma) \qquad \frac{}{XXXXXXXXXXX}\ (\text{Rules for } \Pi)$$

4.2.1. *Interpretation into $\mathbb{T}^w$.* TODO: $\mathbb{T}^{\mathsf{coh}} \to \mathbb{T}^w$

4.2.2. *Coherence.* We now focus on proving the following property of $\mathbb{T}^{\mathsf{coh}}$.

Lemma 3. *For any context* Γ *of* $\mathbb{T}^w$*, seen as a context of* $\mathbb{T}^{\mathsf{coh}}$*, typed term* $(\Gamma \overset{\mathsf{coh}}{\vdash} x : A)$ *in* $\mathbb{T}^{\mathsf{coh}}$*, and loop* $(\Gamma \overset{\mathsf{coh}}{\vdash} p : \mathsf{Coh}(x,x))$*, there is a term* $p^\dagger$ *of type* $\mathsf{Coh}(p, \overset{\mathsf{coh}}{\mathsf{refl}}\,(x))$ *in the context* Γ*.*

As a first approximation, we can sketch a proof of the same result in a type theory built in the same way as $\mathbb{T}^{\mathsf{coh}}$, but without Σ-types, Π-types nor the terms e_{J} and e_{J_β}. Assuming that the context Γ comes from $\mathbb{T}^w$, and thus does not contain coherence types, it is possible to associate to any term $(\Gamma \overset{\mathsf{coh}}{\vdash} x : A)$ a normal form $|x| : |A|$ obtained by erasing all occurences of $\overset{\mathsf{coh}}{\mathsf{J}}$ and $\overset{\mathsf{coh}}{\mathsf{J}}_\beta$. An important property of these normal forms is that if a coherence type $\mathsf{Coh}(x,y)$ in inhabited, then x and y have the same normal forms ($|x| = |y|$). By induction over the syntax, it is then possible to coherently relate all terms to their normal form. We get for any term $x : A$ of a basic type a dependent coherence $[x] : (x : A) \simeq (|x| : |A|)$. For any coherence $p : \mathsf{Coh}(A, x, y)$ in a basic type A, this gives a higher coherence $[p] : \mathsf{Coh}(p, [x]^{-1}; [y])$, witnessing the commutativity of the diagram

(7)
$$
\begin{array}{ccc}
x & \xrightarrow{\;p\;} & y \\
\Big\downarrow{\scriptstyle [x]} & & \Big\downarrow{\scriptstyle [y]} \\
|x| & = \!= \!= & |y|\ .
\end{array}
$$

In the particular case of a loop $p : \mathsf{Coh}(a,a)$, it provides an inhabitant of $\mathsf{Coh}(p, [a]^{-1}; [a])$, from which we can derive a proof of $\mathsf{Coh}(p, \overset{\mathsf{coh}}{\mathsf{refl}}\,(a))$. Also dealing with Σ-types and Π-types should be possible with minor modifications to the proof.

We would like to also extend this proof to our type theory $\mathbb{T}^{\mathsf{coh}}$, that includes the coherences e_{J} and e_{J_β}. However, it seems that defining normal forms in a suitable way is no longer possible. Our solution is to use head normal forms instead. Let us give a simplified outline of the proof, before getting into the details.

We will define for any term $(\Gamma \overset{\mathsf{coh}}{\vdash} x : A)$ a triple $(x^{\mathsf{H}}, \sigma_x, f_x)$ that we see as a decomposition of the head normal form of x. $x^{\mathsf{H}} : \mathsf{Ty}^{\mathsf{coh}}(\diamond)$ is the closed type of parameters of the head symbol of x. $f_x : \mathsf{Ter}((\bar{x} : x^{\mathsf{H}}), -)$ is the head symbol of x, a term whose type we will precise later. $\sigma_x : \mathsf{Ter}(\Gamma, x^{\mathsf{H}})$ are the parameters of the head symbol of x. When x is a term of a basic type in normal form, we will have $x = f_x(\sigma_x)$, i.e. x can be exactly be recovered from its head normal form decomposition. A first example of this decomposition is $\mathsf{refl}(\mathsf{X}, x)^{\mathsf{H}} = \mathsf{X}$, $\sigma_{\mathsf{refl}(\mathsf{X},x)} = x$ and $f_{\mathsf{refl}(\mathsf{X},x)} = ((\bar{x} : \mathsf{X}) \vdash \mathsf{refl}(\mathsf{X}, \bar{x}) : \mathsf{Id}(\mathsf{X}, \bar{x}, \bar{x}))$. Our notion of head symbol includes the structure of types under the head term operation: we have for instance $\mathsf{refl}(\mathsf{Id}(\mathsf{X}, x, y), z) = \Sigma(\bar{x} : \mathsf{X})(\bar{y} : \mathsf{X})(\bar{z} : \mathsf{Id}(\mathsf{X}, \bar{x}, \bar{z}))$ and $f_{\mathsf{refl}(\mathsf{Id}(\mathsf{X},x,y),z)} = (\bar{x}, \bar{y}, \bar{z} \vdash \mathsf{refl}(\bar{z}) : \mathsf{Id}(\mathsf{X}, \bar{x}, \bar{y}))$. Head symbols are the same for terms related by a coherent identity: whenever $\mathsf{Coh}(x,y)$ is derivable, we will have that $(x^{\mathsf{H}}, f_x) = (y^{\mathsf{H}}, f_y)$.

We will define for any type $(\Gamma \overset{\mathsf{coh}}{\vdash} A)$ of $\mathbb{T}^{\mathsf{coh}}$, and any possible head symbol $(x^{\mathsf{H}}, f_x) \in \{(x^{\mathsf{H}}, f_x) \mid \mathsf{Ter}^{\mathsf{coh}}(\Gamma, A)\}$ a type $[A]_{x^{\mathsf{H}}} : \mathsf{Ty}^{\mathsf{coh}}([\Gamma], (a : A), (a_0 : x^{\mathsf{H}}))$ of "coherent relations" between terms and their head symbol (or proofs of local contractibility from a term to their head normal form). At the same time, we define for any term $(\Gamma \overset{\mathsf{coh}}{\vdash} a : A)$ a proof of $\Sigma(a_0 : a^{\mathsf{H}})(a_1 : [A]_{a^{\mathsf{H}}}(a, a_0))$.

This yields the following result for a loop $p : \mathsf{Coh}(A, a, a)$ of a basic type A: there is a coherent equality $q : \mathsf{Coh}(a^{\mathsf{H}}, \sigma_a, \sigma_a)$, and a higher coherence witnessing the commutativity of the diagram

(8)
$$\begin{array}{ccc} a & \xrightarrow{p} & a \\ \Big\downarrow{\scriptstyle [a]} & & \Big\downarrow{\scriptstyle [a]} \\ f_a(\sigma_a) & \xrightarrow{\mathsf{ap}(f_a)(q)} & f_a(\sigma_a) \end{array} .$$

We can proceed recursively by proving that the loop q is trivial. The termination of this procedure is ensured by fact that the terms of $\mathbb{T}^s$ are strongly normalizing.

Replace the word "symbol" ?

Replace the expression "basic type" ?

Remove the variable head symbol types and the context to ρ and τ to deal with variables.

Definition 23 (Head symbols). *A type head symbol is a couple* (H, τ)*, where* H *is a closed type of* $\mathbb{T}^{\mathsf{coh}}$*, and* τ *is a type over* H *in* $\mathsf{Ty}^{\mathsf{coh}}((\bar{h} : \mathsf{H}))$*.*
A term head symbol in a context Γ *over a type head symbol* (H_A, τ_A) *is a triple* (H, ρ, τ)*, where* H *is a closed type,* ρ *is a term of* $\mathsf{Ter}^{\mathsf{coh}}((\bar{h} : \mathsf{H}), \mathsf{H}_A)$*, and* τ *is either a variable of a basic type of* Γ*, or a term in* $\mathsf{Ter}^{\mathsf{coh}}((\bar{h} : \mathsf{H}), \tau_A(\rho(h)))$*.*

Definition 24 (Head normal form decomposition). *We define inductively a head normal form decomposition for every type and term of* $\mathbb{T}^{\mathsf{coh}}$*.*

- *For every type* $(\Gamma \overset{\mathsf{coh}}{\vdash} A\ \mathsf{type})$*, we give a head symbol* $A^{\mathsf{H}} = (\mathsf{H}_A, \tau_A)$*, and when* A *is basic, a term* $\sigma_A \in \mathsf{Ter}^{\mathsf{coh}}(\Gamma, \mathsf{H}_A)$ *such that* $\tau_A \circ \sigma_A \equiv A$*.*
- *For every term* $(\Gamma \overset{\mathsf{coh}}{\vdash} a : A)$*, we give a term head symbol* a^{H}*.*

Definition.
Types:

- $(\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{X}\ \mathsf{basic\ type})$:

$$\mathsf{H} = \mathbf{1}$$
$$\tau = () \mapsto \mathsf{X}$$
$$\sigma = ()$$

- $(\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{Id}(A, x, y)\ \mathsf{basic\ type})$:

$$\mathsf{H} = \Sigma(\bar{A} : \mathsf{H}_A)(\bar{x} : \tau_A(\bar{A}))(\bar{y} : \tau_A(\bar{A}))$$
$$\tau = (\bar{A}, \bar{x}, \bar{y}) \mapsto \mathsf{Id}(\tau_A(\bar{A}), \bar{x}, \bar{y})$$
$$\sigma = (\sigma_A, x, y)$$

- $(\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{Coh}(A, x, y)\ \mathsf{type})$:

$$\mathsf{Coh}(A, x, y)^{\mathsf{H}} = (\mathsf{H}_x, \tau_A \circ \rho_x)$$

- $(\Gamma \overset{\mathsf{coh}}{\vdash} \Sigma AB\ \mathsf{type})$:

$$\mathsf{H} = \Sigma(\bar{A} : \mathsf{H}_A)(\bar{B} : \tau_A(\bar{A}) \to \mathsf{H}_B)$$
$$\tau = (\bar{A}, \bar{B}) \mapsto \Sigma(a : \tau_A(\bar{A}))(\tau_B(\bar{B}(a)))$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \Pi AB$ type):

$$\mathsf{H} = \Sigma(\bar{A} : \mathsf{H}_A)(\bar{B} : \tau_A(\bar{A}) \to \mathsf{H}_B)$$
$$\tau = (\bar{A}, \bar{B}) \mapsto \Pi(a : \tau_A(\bar{A}))(\tau_B(\bar{B}(a)))$$

Terms:

- ($\Gamma \overset{\mathsf{coh}}{\vdash} x_i : A_i$):

$$\mathsf{H} = \mathsf{H}_{A_i}$$
$$\rho = \mathsf{id}$$
$$\tau = x_i$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{refl}(A, x) : \mathsf{Id}(A, x, x)$):

$$\mathsf{H} = \Sigma(\bar{A} : \mathsf{H}_A)(\bar{x} : \tau_A(\bar{A}))$$
$$\rho = (\bar{A}, \bar{x}) \mapsto (\bar{A}, \bar{x}, \bar{x})$$
$$\tau = (\bar{A}, \bar{x}) \mapsto \mathsf{refl}(\tau_A(\bar{A}), \bar{x})$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{J}^{\Delta,C}_{A,x}(d, y, p, \delta)) : C(y, p, \delta)$ when p reduces to refl:

$$\mathsf{J}^{\mathsf{H}} = d^{\mathsf{H}}$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{J}^{\Delta,C}_{A,x}(d, y, p, \delta)) : C(y, p, \delta)$ otherwise:

$$\begin{aligned}\mathsf{H} = \Sigma(\bar{A} : \mathsf{H}_A)(\bar{x} : \tau_A(\bar{A}))(\bar{\Delta} : (\bar{y} : \tau_A(\bar{A}))(\bar{p} : \mathsf{Id}(\bar{x}, \bar{y})) \to \mathsf{H}_\Delta)\\ (\bar{C} : (\bar{y} : \tau_A(\bar{A}))(\bar{p} : \mathsf{Id}(\bar{x}, \bar{y}))(\bar{\delta} : \tau_\Delta(\bar{\Delta}(\bar{y}, \bar{p}))) \to \mathsf{H}_C)\\ (d : (\bar{\delta} : \tau_\Delta(\bar{\Delta}(\bar{x}, \mathsf{refl}(\bar{x})))) \to \tau_C(\bar{C}(\bar{x}, \mathsf{refl}(\bar{x}), \bar{\delta})))\\ (\bar{y} : \tau_A(\bar{A}))(\bar{p} : \mathsf{Id}(\bar{x}, \bar{y}))(\bar{\delta} : \tau_\Delta(\bar{\Delta}(\bar{y}, \bar{p})))\end{aligned}$$
$$\rho = (\dots) \mapsto \bar{C}(\bar{y}, \bar{p}, \bar{\delta})$$
$$\tau = (\dots) \mapsto \mathsf{J}^{\tau_\Delta(\bar{\Delta}), \tau_C(\bar{C})}_{\tau_A(\bar{A}), \bar{x}}(\bar{d}, \bar{y}, \bar{p}, \bar{\delta})$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{J}^{\Delta,C}_{\beta,A,x}(d, \delta) : \mathsf{Id}(\mathsf{J}(d, x, \mathsf{refl}(x), \delta), d(\delta)))$:

$$\mathsf{J}^{\mathsf{H}}_\beta = \mathsf{refl}(d)^{\mathsf{H}}$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \overset{\mathsf{coh}}{\mathsf{refl}}(A, x) : \mathsf{Coh}(x, x)$):

$$\overset{\mathsf{coh}}{\mathsf{refl}}{}^{\mathsf{H}} = x^{\mathsf{H}}$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \overset{\mathsf{coh}}{\mathsf{J}}(d, y, p, \delta)$):

$$\overset{\mathsf{coh}}{\mathsf{J}}{}^{\mathsf{H}} = d^{\mathsf{H}}$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \overset{\mathsf{coh}}{\mathsf{J}}_\beta(d, \delta)$):

$$\overset{\mathsf{coh}}{\mathsf{J}}_\beta = d^{\mathsf{H}}$$

- $(\Gamma \overset{\mathsf{coh}}{\vdash} e_{\mathsf{J}}(d, \delta))$:
$$e_{\mathsf{J}}^{\mathsf{H}} = d^{\mathsf{H}}$$
- $(\Gamma \overset{\mathsf{coh}}{\vdash} e_{\mathsf{J}_\beta}(d, \delta))$:
$$e_{\mathsf{J}_\beta}^{\mathsf{H}} = d^{\mathsf{H}}$$
- $\lambda, \mathsf{app}, \pi_1, \pi_2, \mathsf{pair}, \ldots$: TODO

□

Proposition 9. *Let* $(\Gamma \overset{\mathsf{coh}}{\vdash} p : \mathsf{Coh}(A, x, y)$ *be a coherent identity in* $\mathbb{T}^{\mathsf{coh}}$. *Then* x *and* y *have the same head symbol:*
$$x^{\mathsf{H}} = y^{\mathsf{H}}.$$

Proposition 10. *other properties of the decompositions...*

Definition 25. *For any type* $(\Gamma \overset{\mathsf{coh}}{\vdash} A)$, *we denote by* $\mathcal{H}(A)$ *the set of possible term head symbols at* A*:*
$$\mathcal{H}(A) = \{x^{\mathsf{H}} \mid x \in \mathsf{Ter}^{\mathsf{coh}}(\Gamma, A)\}.$$
For any context $\Gamma = (A_0, A_1, \ldots, A_{n-1})$ *in* $\mathsf{Con}^{\mathsf{coh}}$, *we denote by* $\mathcal{H}(\Gamma)$ *the product*
$$\mathcal{H}(\Gamma) = \prod_{i<n} \mathcal{H}(A_0, A_1, \ldots, A_i \overset{\mathsf{coh}}{\vdash} A_i).$$
Note that the type A_i *is repeated in the context, so as to allow for variable term head symbols.*

Definition 26 ((Logical relations/Parametricity) Translation)**.** *We define:*

- *For any context* $\Gamma = (A_0, A_1, \ldots, A_{n-1})$ *in* $\mathsf{Con}^{\mathsf{coh}}$, *and term head symbols* $h \in \mathcal{H}(\Gamma)$, *a context* $[\Gamma]_h$ *which is a weakening of* Γ,
- *For any type* $(\Gamma \overset{\mathsf{coh}}{\vdash} A)$, *and any term head symbol* $h \in \mathcal{H}(A)$, *a type*
$$[A]_h \in \mathsf{Ty}([\Gamma], (a : A), (a_0 : \mathsf{H}_h)),\ and$$
- *For any term* $(\Gamma \overset{\mathsf{coh}}{\vdash} a \in A)$, *terms*
$$[a]_0 \in \mathsf{Ter}^{\mathsf{coh}}([\Gamma], \mathsf{H}_a),\ and$$
$$[a]_1 \in \mathsf{Ter}^{\mathsf{coh}}([\Gamma], [A]_{a^{\mathsf{H}}}(a, [a]_0)).$$

Definition.
Contexts:

- $(\diamond \vdash)$:
$$[\diamond]_{()} = \diamond.$$
- $(\Gamma, (a : A) \vdash)$:
$$[\Gamma, (a : A)]_{(h, h_a)} = [\Gamma]_h, (a : A), (a_0 : \mathsf{H}_{h_a}), (a_1 : [A]_{h_a}(a, a_0)).$$

Types:

- ($\Gamma \overset{\mathsf{coh}}{\vdash} A$ basic type):
  Let $h \in \mathcal{H}(A)$ a term head symbol at A, and work in the context $[\Gamma], (a : A), (a_0 : \mathsf{H}_h)$. Note that since A is basic, we have $A \equiv f_A(\sigma_A)$.

  This doesn't work when τ_h is a variable.

  We would like to define $[A]$ to be the type of coherent equalities between $a : A$ and $\tau_h(a_0) : f_A(\rho_h(a_0))$. Because these terms live in different types, we additionally need an equivalence between the two types $f_A(\sigma_A)$ and $f_A(\rho_h(a_0))$. As such an equivalence can be obtained from any coherent equality between σ_A and $\rho_h(a_0)$, when τ_h is not a variable, we define $[A]$ to be the type

$$
\begin{aligned}
[A]_h(a, a_0) = \Sigma(&q : \mathsf{Coh}(\rho_h(a_0), \sigma_A))\\
&(\mathsf{Coh}(A, a, \mathsf{tr}_q^{f_A}(\tau_h(a_0)))).
\end{aligned}
$$

  When τ_h is a variable x_i of a basic type A_i in Γ, we have that $A^{\mathsf{H}} = A_i^{\mathsf{H}}$, and we can define instead:

$$
\begin{aligned}
[A]_h(a, a_0) = \Sigma(&q_0 : \mathsf{Coh}(\rho_h(a_0), \sigma_A))\\
&(q_1 : \mathsf{Coh}(\sigma_{A_i}, \sigma_A))\\
&(\mathsf{Coh}(A, a, \mathsf{tr}_{q_1}^{f_A}(x_i))).
\end{aligned}
$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{Coh}(A, x, y)$ type):
  By (proposition ...), the only term head symbol at the coherence type $\mathsf{Coh}(A, x, y)$ is x^{H}.

$$
\begin{aligned}
[\mathsf{Coh}(A, x, y)]_h(a, a_0) = \Sigma(&q_x : \mathsf{Coh}([x]_0, a_0))\\
&(q_y : \mathsf{Coh}([y]_0, q_0))\\
&(\mathsf{Coh}([A](x, a_0), \mathsf{tr}_{q_x}^{[A](x,-)}([x]_1), \mathsf{tr}_a^{[A](-,a_0)}(\mathsf{tr}_{q_y}^{[A](y,-)}([y]_1)))).
\end{aligned}
$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \Sigma AB$ type):
  TODO
- ($\Gamma \overset{\mathsf{coh}}{\vdash} \Pi AB$ type):
  TODO

Terms:

- ($\Gamma \overset{\mathsf{coh}}{\vdash} x_i : A_i$):

$$
[x_i]_0 = \sigma_{A_i}
$$
$$
[x_i]_1 = (\overset{\mathsf{coh}}{\mathsf{refl}}, \overset{\mathsf{coh}}{\mathsf{refl}}, \mathsf{tr}_\beta)
$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{refl}(A, x) : \mathsf{Id}(A, x, x)$):

$$
[\mathsf{refl}(A, x)]_0 = (\sigma_A, x)
$$
$$
[\mathsf{refl}(A, x)]_1 = (\overset{\mathsf{coh}}{\mathsf{refl}}, \mathsf{tr}_\beta)
$$

- ($\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{J}_{A,x}^{\Delta,C}(d, y, p, \delta)) : C(y, p, \delta)$ when p reduces to refl:

- $(\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{J}^{\Delta,C}_{A,x}(d,y,p,\delta)) : C(y,p,\delta)$ otherwise:

$$[\mathsf{J}]_0 = (\sigma_A, x, \sigma_\Delta, \sigma_C, d, y, p, \delta)$$
$$[\mathsf{J}]_1 = (\overset{\mathsf{coh}}{\mathsf{refl}}, \mathsf{tr}_\beta)$$

- $(\Gamma \overset{\mathsf{coh}}{\vdash} \mathsf{J}^{\Delta,C}_{\beta,A,x}(d,\delta) : \mathsf{Id}(\mathsf{J}(d,x,\mathsf{refl}(x),\delta), d(\delta)))$:

- $(\Gamma \overset{\mathsf{coh}}{\vdash} \overset{\mathsf{coh}}{\mathsf{refl}}(A,x) : \mathsf{Coh}(x,x))$:

$$[\overset{\mathsf{coh}}{\mathsf{refl}}(A,x)]_0 = [x]_0$$
$$[\overset{\mathsf{coh}}{\mathsf{refl}}(A,x)]_1 = (\overset{\mathsf{coh}}{\mathsf{refl}}, \overset{\mathsf{coh}}{\mathsf{refl}}, \mathsf{tr}_\beta)$$

- $(\Gamma \overset{\mathsf{coh}}{\vdash} \overset{\mathsf{coh}}{\mathsf{J}}(d,y,p,\delta))$:

- $(\Gamma \overset{\mathsf{coh}}{\vdash} \overset{\mathsf{coh}}{\mathsf{J}}_\beta(d,\delta))$:

- $(\Gamma \overset{\mathsf{coh}}{\vdash} e_{\mathsf{J}}(d,\delta))$:

- $(\Gamma \overset{\mathsf{coh}}{\vdash} e_{\mathsf{J}_\beta}(d,\delta))$:

- $\lambda, \mathsf{app}, \pi_1, \pi_2, \mathsf{pair}, \ldots$: TODO

□

Proposition 11. *Let $\Gamma \in \mathsf{Con}^w$ be a context of $\mathbb{T}^w$, and A be a basic type of $\mathbb{T}^{\mathsf{coh}}$ in $\mathsf{Ty}^{\mathsf{coh}}(\Gamma)$. Take $a \in \mathsf{Ter}^{\mathsf{coh}}(\Gamma, A)$ and $p \in \mathsf{Ter}^{\mathsf{coh}}(\Gamma, \mathsf{Coh}(A,a,a))$.*
Then we can construct:

- *A type equivalence (using the coherent identity types) $\Gamma \overset{\mathsf{coh}}{\vdash} w : A \simeq f_A(p_a(\sigma_a))$,*
- *A term $\hat{a} \in \mathsf{Ter}^{\mathsf{coh}}(\Gamma, \mathsf{Coh}(w(a), f_a(\sigma_a)))$,*
- *A term $q \in \mathsf{Ter}^{\mathsf{coh}}(\Gamma, \mathsf{Coh}(a^{\mathsf{H}}, \sigma_a, \sigma_a))$,*
- *A term $\hat{p} \in \mathsf{Ter}^{\mathsf{coh}}(\Gamma, \mathsf{Coh}(\hat{a}; \mathsf{ap}(f_a)(\sigma_a), p; \hat{a}))$.*

Proof. TODO □

Proof of lemma 3. TODO
main idea : induction on the normal forms of $\mathbb{T}^s$, and repeated applications of the previous proposition. □

4.3. **Conservativity.** Finally, we use the type theory $\mathbb{T}^{\mathsf{coh}}$ we have just defined in order to construct equivalences in a coherent manner, and thus prove the conservativity of $\mathbb{T}^s$ over $\mathbb{T}^w$.

Lemma 4. *We prove by induction over the equivalence relations defined in definition 21 that:*

- *for any pair of related contexts $\Gamma \sim \Delta$, there is an equivalence $w : \Gamma \simeq \Delta$ in $\mathbb{T}^{\mathsf{coh}}$,*
- *for any pair of related types $(\Gamma \overset{w}{\vdash} A) \sim (\Delta \overset{w}{\vdash} B)$, there is a dependent equivalence $f : A \simeq B$ in $\mathbb{T}^{\mathsf{coh}}$, and*
- *for any pair of related terms $(\Gamma \overset{w}{\vdash} a : A) \sim (\Delta \overset{w}{\vdash} b : B)$, there is a dependent coherent equality $p : a \simeq b$ in $\mathbb{T}^{\mathsf{coh}}$.*

Proof. TODO, using lemma 3 □

Theorem 3. *$\mathbb{T}^s$ is conservative over $\mathbb{T}^w$.*

TODO