HOMOTOPICAL STRUCTURES IN DEPENDENT TYPE THEORY

MARCO VERGURA

Contents

Introduction

There is a compelling hermeneutical prejudice that affects probably every newcomer to the study of (dependent) type theory, namely the idea that types should be read as sets and terms of a type should be read as elements of a set. Such a natural interpretation is however doomed to crumple as soon as it tries to make sense of identity types: given two elements a, a' of a set A, the only way we can compare (or "connect") them is by asking whether or not they are equal. Thus, a set that had to correspond to the identity type $\mathrm{Id}_A(a, a)$ could only be the empty set or a singleton set, making identity types essentially trivial.

It is precisely in this respect that the homotopical interpretation of type theory shows all his fascinating power. Indeed, if we think to types (or to contexts) as *spaces* rather than sets and to terms of a type as points of such a space, then, given two points a, a' of a space A, we have indeed a whole space of connections between these two points, namely, the space of all paths in A from a to a'. We can even patch together all these spaces of paths with fixed endpoints as fibers of a fibration $A^I \to A \times A$ of spaces and there is a canonical homotopy equivalence $A \to A^I$ sending $a : A$ into the constant path at a (which provides the canonical inhabitant of $\mathrm{Id}_A(a, a)$). The outstanding aspect of this viewpoint is that, for some reasons that we can not still grasp fully, it does not end in being a mere exercise in style but it seems to give indeed a lot of insights to some structures that naturally arise in type theory. The goal of the present work is to try and explore some of these homotopical built-ins of dependent type theory.

Date: December 10, 2015.

The first step to make in order to talk about homotopical perspectives in type theory is to organize the syntactic data of a dependent type theory $\mathbb{T}$ satisfying the structural rules for judgments and for definitional equality into a category, the *classifying category* $\mathcal{C}\ell(\mathbb{T})$ of $\mathbb{T}$. The objects of such a category are contexts (up to definitional equality) and the maps are context morphisms (up to definitional equality). Among these morphisms, *dependent projections* $\pi_\Phi\colon [\Gamma,\ \Phi] \to \Gamma$ are of particular relevance as they allow, for example, to pullback maps in $\mathcal{C}\ell(\mathbb{T})$ along them and to define *(dependent) terms* $a{:}\Phi$ of a context Φ. The key idea now is that contexts (rather than types) should behave as spaces. Indeed, if our type theory $\mathbb{T}$ has identity types, we can define *identity contexts*: given $x, x'{:}\Phi$ for a context Φ, we can obtain a new context $\mathrm{Id}_\Phi(x, x')$ and such contexts satisfy introduction, elimination and computation rules analogous to those for identity types. Terms of identity contexts $\mathrm{Id}_\Phi(x, x')$ are called *paths* (from x to x') and paths between paths are called *homotopies*. These names are totally legitimate as we can compose paths and such a composition operation is associative, unital and has inverses up to homotopy. This allows us to define a functor

$$\Pi_1\colon \mathcal{C}\ell(\mathbb{T}) \longrightarrow \mathsf{Gpd}$$

which associates to each context Φ its *fundamental groupoid*: this construction is the formal analogue to the one defining the fundamental groupoid of a space.

The fundamental groupoid construction already gives a flavour of the inherently homotopical nature of a dependent type theory $\mathbb{T}$, but we can do much better if we require $\mathbb{T}$ to have rules for *mapping cylinders for contexts*. These are axioms which allow us to derive, given a context morphism $f\colon \Phi \to \Psi$ and $y : \Psi$, a context $\mathsf{Cyl}_f(y)$ with suitable constructors and computation rules which make Cyl an example of a *higher inductive context*. In the presence of these mapping cylinders, we can show, following [Lum11] and [GG08], that $\mathcal{C}\ell(\mathbb{T})$ carries a (pre-)model category structure, where:

- a map $f\colon \Phi \to \Psi$ is a *weak equivalence* if it has an inverse up to homotopy;
- *fibrations* are generated by dependent projections;
- *cofibrations* are obtained from dependent projections from contractible dependent contexts.

It follows in particular that every object is fibrant in such a model category structure. This is probably one of the most precise way in which we can say that there is a homotopical content in dependent type theory: $\mathcal{C}\ell(\mathbb{T})$, being a (pre-)model category, is a presentation for a homotopy theory, namely its associated homotopy category!

Although we can not avoid using mapping cylinders if we want to get the full pre-model category structure above, they are crucially needed only to show that each map in $\mathcal{C}\ell(\mathbb{T})$ factors as a cofibration followed by a trivial fibration. Indeed, the factorization of each map into a trivial cofibration followed by a fibration uses solely the fact that $\mathbb{T}$ has identity types. Namely, identity contexts can be used to factor every context morphism $f\colon \Phi \to \Psi$ into a map $X \to \mathrm{Id}(f) = [x : \Phi,\ y{:}\Psi,\ u{:}\mathrm{Id}(fx, y)]$ followed by a dependent projection. This is the syntactic counterpart of the factorization of each map of spaces into a homotopy equivalence followed by a fibration and it allows us to show that the trivial cofibrations and the fibrations form a weak factorization system on $\mathcal{C}\ell(\mathbb{T})$, even when just identity types are available for $\mathbb{T}$.

The existence of a remarkable amount of homotopical information built into the classifying category of any dependent type theory $\mathbb{T}$ admitting only identity types suggests us to look for an abstract categorical structure which, albeit weaker than the concept of a model category, still admits enough structure to capture the interesting properties of $\mathcal{C}\ell(\mathbb{T})$. This task is partly accomplished via the notion of *fibration category* as we can prove (following [AKL15]) that the class of equivalences and the class of dependent projections (once they are closed under isomorphisms) turns

$\mathcal{C}\ell(\mathbb{T})$ into a fibration category.

Our work is organized as follow. In Section 1, we explain our type-theoretic setting for the rest of the paper and introduce identity contexts and mapping cylinders for contexts. Section 2 describes the construction of the classifying category of a dependent type theory $\mathbb{T}$. We also define there dependent projections and display maps which will play a preeminent role throughout our work. In Section 3 we exploit the syntactic power of identity contexts to functorially associate to each context a groupoid (its fundamental groupoid). Section 4 is devoted to the purpose of defining all the classes of maps we will need in order to construct the pre-model structure on the classifying category of a dependent type theory with identity types and mapping cylinders. We prove the existence of such a pre-model category structure in Section 5 by checking in detail all the needed axioms. In doing so, we will realize that most of the homotopical structure present in $\mathcal{C}\ell(\mathbb{T})$ is actually due to identity contexts, which provide a weak factorization system on the classifying category. We close our work in Section 6 where we discuss briefly how, even when mapping cylinders are not available, we can turn $\mathcal{C}\ell(\mathbb{T})$ into a fibration category, by relying only on the presence of identity contexts.

1. Basic rules of Type Theory and the Identity Context

We set here the framework in which we will work from now on. We are going to use a somehow minimalist setting for the dependent type theories we will consider. As usual, we have four basic judgments

$$\Gamma \vdash A : \mathsf{Type}, \quad \Gamma \vdash a : A, \quad \Gamma \vdash A = B : \mathsf{Type}, \quad \Gamma \vdash a = b : A, \tag{1}$$

which express, respectively, that, in the context Γ, A is a *type*, a is a *term of type* A, the types A and B are equal and, finally, a and b are equal as terms of type A. Here and in the following, when we mention *contexts* (which are a finite list of dependent types), we always implicitely assume that they are *well-formed* (i.e. each dependent type in the list forming a context is only allowed to depend upon the previous types in the list). We use the notation

$$\vdash \Gamma : \mathsf{Cxt}$$

to mean that the context Γ is well-formed. Thus, technically speaking, all our basic judgments in (1) should be preceeded by the judgment " $\vdash \Gamma : \mathsf{Cxt}$". We require that contexts verify the following rules, where Γ and Δ are contexts and $\mathcal{J}$ is an arbitrary judgment:

$$\frac{}{\Gamma,\ x : A \vdash x : A}\ \textsc{Vble} \qquad \frac{\Gamma,\ \Delta \vdash \mathcal{J} \qquad \Gamma \vdash A : \mathsf{Type}}{\Gamma,\ x : A,\ \Delta \vdash \mathcal{J}}\ \textsc{Wkg} \qquad \frac{\Gamma,\ x : A,\ \Delta \vdash \mathcal{J} \qquad \Gamma \vdash a : A}{\Gamma,\ \Delta[a/x] \vdash \mathcal{J}[a/x]}\ \textsc{Sub} \tag{2}$$

Here and in the following, the equality symbol is always used to mean *definitional equality* (or *judgmental equality*), which is subject to the following structural rules:

$$\begin{gathered}
\frac{\Gamma \vdash A : \mathsf{Type}}{\Gamma \vdash A = A : \mathsf{Type}} \qquad \frac{\Gamma \vdash A = B : \mathsf{Type}}{\Gamma \vdash B = A : \mathsf{Type}} \qquad \frac{\Gamma \vdash A = B : \mathsf{Type} \qquad \Gamma \vdash B = C : \mathsf{Type}}{\Gamma \vdash A = C : \mathsf{Type}} \\
\frac{\Gamma \vdash a : A}{\Gamma \vdash a = a : A} \qquad \frac{\Gamma \vdash a = b : \mathsf{Type}}{\Gamma \vdash b = a : A} \qquad \frac{\Gamma \vdash a = b : A \qquad \Gamma \vdash b = c : A}{\Gamma \vdash a = c : A} \\
\frac{\Gamma \vdash a : A \qquad \Gamma \vdash A = B : \mathsf{Type}}{\Gamma \vdash a : B} \qquad \frac{\Gamma \vdash a = b : A \qquad \Gamma \vdash A = B : \mathsf{Type}}{\Gamma \vdash a = b : B}
\end{gathered} \tag{3}$$

For a large part of our work, the only type constructor we will require our type theories to have is given by identity types.

Definition 1.1. [GG08, Section 2, Table 1] & [Gar09, Section 2]. We introduce the *identity type* family via the following rules.

$$\frac{\Gamma \vdash A:\mathsf{Type} \qquad \Gamma \vdash a,b:A}{\Gamma \vdash \mathrm{Id}_A(a,b):\mathsf{Type}}\ \text{Id-FORM} \qquad \frac{\Gamma \vdash A:\mathsf{Type} \qquad \Gamma \vdash a:A}{\Gamma \vdash r_A(a):\mathrm{Id}_A(a,a)}\ \text{Id-INTRO}$$

(4)
$$\frac{\begin{array}{c}\Gamma,\ x,y:A,\ u:\mathrm{Id}_A(x,y),\ \Delta \vdash C(x,y,u):\mathsf{Type}\\ \Gamma,\ z:A,\ \Delta[z,z/y,r_A(z)/u] \vdash d(z) : C(z,z,r_A(z))\end{array}}{\Gamma,\ x,y:A,\ u:\mathrm{Id}_A(x,y),\ \Delta \vdash J_d(x,y,u) : C(x,y,u)}\ \text{Id-ELIM}$$

$$\frac{\begin{array}{c}\Gamma,\ x,y:A,\ u:\mathrm{Id}_A(x,y),\ \Delta \vdash C(x,y,u):\mathsf{Type}\\ \Gamma,\ z:A,\ \Delta[z,z/y,r_A(z)/u] \vdash d(z) : C(z,z,r_A(z))\end{array}}{\Gamma,\ x:A,\ \Delta[x,x/y,r_A(x)/u] \vdash J_d(x,x,r_A(x)) = d(x) : C(x,x,r_A(x))}\ \text{Id-COMP}$$

(In Id $-$ELIM and Id $-$INTRO the context Δ is a dependent context relative to $[\Gamma,\ x:A,\ y:A,\ u:\mathrm{Id}_A(x,y)]$ (see Definition 1.4)).

Remark 1.2. Note the presence of the additional context Δ in the computation and in the elimination rule for the identity type. Such an extended version of these rules (with respect, for example, to the account given in [Pro13, Appendix A]; see also [Gar09, Section2, Table 1]) is needed because we are not assuming our dependent type theories $\mathbb{T}$ to have Σ and $\Pi-$types. However, in the presence of dependent sums and dependent products, the Id $-$ELIM and Id $-$COMP rules in (4) are derivable from a formulation of the identity types where the elimination and the computation rules do not include the extra context Δ. The reason is essentially the following. Given a context

$$\Delta = [w_0:T_0,\ \ldots,\ w_k:T_k(w_1,\ \ldots,\ w_{k-1})],$$

our default assumption on contexts being well-formed and the formation rule for dependent sums allow us to derive the judgment

$$\vdash \sum_{w_0:\ T_0,\ \ldots,\ w_{k-1}:\ T_{k-1}} T_k,$$

where the sum over $w_0,\ \ldots,\ w_{k-1}$ has to be intended as a reiterated sum over the single variables. (For easiness of exposition, we are assuming Δ to be a context which is not dependent relative to another one, but nothing would change if we allowed such a higher level of generality). For example, if $\Delta = [w_0:T_0,\ w_1:T_1]$, then $\vdash \Delta:\mathsf{Cxt}$ means that the judgments

$$\vdash T_0:\mathsf{Type} \quad \text{and} \quad w_0:T_0 \vdash w_1:T_1$$

are derivable. But these are exactly the premises for the formation of the dependent sum $\sum_{w_0:\ T_0} T_1$, so that we can infer

$$\vdash \sum_{w_0:\ T_0} T_1:\mathsf{Type}.$$

In other words, Σ-types allow us to syntactically reinterpret contexts as actual types. Now, if we are given the premises

$$\Gamma,\ x,y:A,\ u:\mathrm{Id}_A(x,y),\ \Delta \vdash C(x,y,u):\mathsf{Type} \qquad \Gamma,\ z:A,\ \Delta[z,z/y,r_A(z)/u] \vdash d(z) : C(z,z,r_A(z))$$

of the Id-ELIM in (4), then, by considering Δ as the associated $(\Sigma-)$ type, the formation rules for Π-types give us

$$\Gamma,\ x,y:A,\ u:\mathrm{Id}_A(x,y) \vdash \prod_{w:\ \Delta} C(x,y,u):\mathsf{Type}$$
$$\Gamma,\ z:A \vdash d(z) : \prod_{w[z,z/y,r_A(z)/u]:\ \Delta[z,z/y,r_A(z)/u]} C(z,z,r_A(z))$$

Then the "$\Delta-$free" version of the elimination rule for identity types gives

$$\Gamma,\ x,y:A,\ p:\mathrm{Id}_A(x,y) \vdash J'_d(x,y,p):\prod_{w:\ \Delta} C(x,y,u).$$

Equivalently, we get the judgment

$$\Gamma,\ x,y:A,\ p:\mathrm{Id}_A(x,y),\ w:\Delta \vdash \left(J'_d(x,y,p)\right)(w):C(x,y,u)$$

which gives the conclusion needed for Id −ELIM in (4). The fact that also Id −COMP follows is now a consequence of the "Δ−free" version of the computation rule for identity types.

Blanket Assumption 1.3. *From now on, we suppose to work with a fixed dependent type theory $\mathbb{T}$ which satisfies the basic syntactic axioms of* (2) *and* (3) *together with the rules* (4) *for identity types.*

Our next goal is to get contextual versions of all the axioms (2), (3) and (4).

Definition 1.4. Let Γ be a context. A *dependent context relative to* Γ is a sequence of typed variables

$$\Phi = [x_0 : A_0, \ \dots, \ x_n : A_n(x_0, \ \dots, \ x_{n-1})]$$

such that the following judgments are derivable

$$\Gamma \vdash A_0 : \mathsf{Type}$$

and, for each $i \in \{1, \dots, n\}$,

$$\Gamma, x_0 : A_0, \ \dots, x_{i-1} : A_{i-1} \vdash A_i(x_0, \ \dots, \ x_{i-1}) : \mathsf{Type}.$$

In other words, Φ is a dependent context relative to Γ if and only $\vdash [\Gamma, \ \Phi]$ Cxt, where $[\Gamma, \ \Phi]$ denotes the sequence of variables obtained by concatenating the typed variables in Φ after the typed variables in Γ. We write $\Gamma \vdash \Phi : \mathsf{Cxt}$ to mean that Φ is a dependent context relative to Γ.

Dependent contexts relative to other contexts play the same role as dependent types relative to other types. We can push the analogy further via the following

Definition 1.5. Let Γ be a context and let

$$\Phi = [x_0 : A_0, \ \dots, \ x_n : A_n(x_0, \ \dots, \ x_{n-1})]$$

be a dependent context relative to Γ. A *dependent term of* Φ *relative to* Γ is a sequence of terms

$$a = (a_0 : \ A_0, \ \dots, \ a_n : A_n(a_0, \ \dots, \ a_{n-1}))$$

such that, for each $i \in \{0, \ \dots, \ n\}$, $\Gamma \vdash a_i : A_i(a_0, \ \dots, \ a_{i-1})$. We write $\Gamma \vdash a : \Phi$ to mean that a is a dependent term of Φ relative to Γ.

Thus, we have defined two new kinds of judgments involving contexts, namely $\Gamma \vdash \Phi : \mathsf{Cxt}$ and $\Gamma \vdash a : \Phi$ (where the latter judgment actually assumes the former and both requires that $\vdash \Gamma : \mathsf{Cxt}$). We can also give a meaning to equality of (dependent) contexts and of (dependent) terms of a context

Definition 1.6. Suppose given two dependent contexts

$$\Phi = [x_0 : A_0, \ \dots, x_n : A_n(x_0, \ \dots, \ x_{n-1})]$$

and

$$\Psi = [y_0 : B_0, \ \dots, y_m : B_m(y_0, \ \dots, \ y_{m-1})]$$

relative to the context Γ.

(1) We say that Φ and Ψ are *(definitionally) equal* and we write $\Gamma \vdash \Phi = \Psi : \mathsf{Cxt}$, if they are componentwise definitionally equal, i.e. $m = n$ and the following judgments are derivable:

$$\Gamma \vdash A_0 = B_0 : \mathsf{Type}$$

and, for all $i \in \{1, \ \dots, \ m\}$,

$$\Gamma, \ [x_0 : A_0, \ \dots, \ x_{i-1} : A_{i-1}], \ [y_0 : B_0, \ \dots, \ y_{i-1} : B_{i-1}] \vdash A_i = B_i : \mathsf{Type},$$

where we wrote A_i and B_i instead of $A_i(x_0, \ \dots, \ x_{i-1})$ and $B_i(y_0, \ \dots, \ y_{i-1})$ respectively.

(2) If $\Gamma \vdash a, b : \Phi$ with $a = (a_0, \ \dots, \ a_n)$ and $b = (b_0, \ \dots, \ b_m)$, we say that a and b are *(definitionally) equal* and we write $\Gamma \vdash a = b : \Phi$, if they are componentwise definitionally equally, i.e. $m = n$ and, for all $j \in \{0, \ \dots, \ m\}$ the following judgment is derivable:

$$\Gamma \vdash a_i = b_i : A_i.$$

All in all, we thus defined the following four judgments

$$\Gamma \vdash \Phi : \mathsf{Cxt}, \quad \Gamma \vdash a : \Phi, \quad \Gamma \vdash \Phi = \Psi : \mathsf{Cxt}, \quad \Gamma \vdash a = b : \Phi, \tag{5}$$

which are the contextual version of the basic judgments in (1). In fact, by substituting the occurrences of each of the basic judgments with their contextual counterparts, we can derive the analogous judgments of (2) and (3) for dependent contexts and dependent elements. The fact that these contextual versions of those judgments hold follows straightforwardly from the definitions of the judgments in (5), which are all given in terms of the four basic judgments in (1), and from the judgments (2) and (3).

We will now show how to use the identity type rules to get analogous rules for an *identity context*. Before doing this, we need a way to pullback terms over paths.

Proposition 1.7. [Gar09, Proposition 3.3.2] *Suppose given a context* Γ *and judgments*

$$\Gamma \vdash A : \mathsf{Type}, \quad \Gamma,\ x : A \vdash B(x) : \mathsf{Type}.$$

Then, the following judgments are derivable in $\mathbb{T}$*:*

$$\frac{\Gamma \vdash a_1, a_2 : A \qquad \Gamma \vdash p : \mathrm{Id}_A(a_1, a_2) \qquad \Gamma \vdash b_2 : B(a_2)}{\Gamma \vdash p^*(b_2) : B(a_1)} *\text{-ELIM}$$
$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash b : B(a)}{\Gamma \vdash (r_A)^*(b) = b : B(a)} *\text{-COMP} \tag{6}$$

Proof. Using Id-ELIM (and weakening for contexts), we get

$$\frac{\Gamma,\ x, y : A,\ u : \mathrm{Id}_A(x, y),\ w : B(y) \vdash B(x) : \mathsf{Type} \qquad \Gamma,\ z : A,\ w : B(z) \vdash w : B(z)}{\Gamma,\ x, y : A,\ u : \mathrm{Id}_A(x, y), w : B(y) \vdash J_{z.w}(x, y, u) : B(x)}$$

and we can set, given the premises of $*-$ELIM, $p^*(b_2) := J_{z.w}(a_1/x, a_2/y, p/u, w/b_2)$. The judgment in $*-$COMP follows now from the computation rule for $J_{z.w}(x, y, u)$. ☐

Theorem 1.8. [Gar09, Proposition 3.3.1]. *The following judgments are derivable in* $\mathbb{T}$*:*

$$\frac{\Gamma \vdash \Phi : \mathsf{Cxt} \qquad \Gamma \vdash a, b : \Phi}{\Gamma \vdash \mathrm{Id}_\Phi(a, b) : \mathsf{Cxt}} \text{Id Cxt-FORM} \qquad \frac{\Gamma \vdash \Phi : \mathsf{Cxt} \qquad \Gamma \vdash a : \Phi}{\Gamma \vdash r_\Phi(a) : \mathrm{Id}_\Phi(a, a)} \text{Id Cxt-INTRO}$$
$$\frac{\begin{array}{c}\Gamma,\ x, y : \Phi,\ p : \mathrm{Id}_\Phi(x, y),\ \Delta \vdash \Theta(x, y, p) : \mathsf{Cxt} \\ \Gamma,\ x : \Phi,\ \Delta[x, x/y, r_\Phi(x)/u] \vdash d(x) : \Theta(x, x, r_\Phi(x))\end{array}}{\Gamma,\ x, y : \Phi,\ p : \mathrm{Id}_\Phi(x, y),\ \Delta \vdash J_d(x, y, p) : \Theta(x, y, p)} \text{Id Cxt-ELIM}$$
$$\frac{\begin{array}{c}\Gamma,\ x, y : \Phi,\ p : \mathrm{Id}_\Phi(x, y),\ \Delta \vdash \Theta(x, y, p) : \mathsf{Cxt} \\ \Gamma,\ x : \Phi,\ \Delta[x, x/y, r_\Phi(x)/p] \vdash d(x) : \Theta(x, x, r_\Phi(x))\end{array}}{\Gamma,\ x : \Phi,\ \Delta[x, x/y, r_\Phi(x)/p] \vdash J_d(x, x, r_\Phi(x)) = d(x) : \Theta(x, x, r_\Phi(x))} \text{Id Cxt-COMP} \tag{7}$$

Proof. We are going to prove (7) by removing the ambient context Γ (which is common to all the premises and the conclusions of the inference rules). Since our proof relies on the rules in (4) for the identity type, which are valid in presence of a background context Γ, it will apply verbatim to show that (7) holds even when Γ is not the empty context, so that our reduction to the case $\Gamma = [\,]$ will cause no loss of generality. In our proofs below, we will also confine ourselves to the case where Δ is the empty context: the same ideas we will exploit can be applied to get the general result.
The proof goes by induction on the length n of the context Φ. In the case $\Phi = [\,]$ is the empty context, we can set $\mathrm{Id}_{[\,]}$ to be the empty context as well. The introduction rule Id Cxt$-$INTRO is vacuously true, whereas Id Cxt$-$COMP and Id Cxt$-$ELIM follow trivially. If $\Phi = [x : A]$ is a

context of length 1, as announced above, we are going to prove that the following judgments are derivable in $\mathbb{T}$:

$$\frac{\vdash \Phi : \mathsf{Cxt} \qquad \vdash a, b : \Phi}{\vdash \mathrm{Id}_\Phi(a,b) : \mathsf{Cxt}} \qquad \frac{\vdash \Phi : \mathsf{Cxt} \qquad \vdash a : \mathsf{Cxt}}{\vdash r_\Phi(a) : \mathrm{Id}_\Phi(a,a)}$$

$$(8) \qquad \frac{x,y:\Phi,\ p:\mathrm{Id}_\Phi(x,y) \vdash \Theta(x,y,p):\mathsf{Cxt} \qquad x:\Phi \vdash d(x) : \Theta(x,x,r_\Phi(x))}{x,y:\Phi,\ p:\mathrm{Id}_\Phi(x,y), \vdash J_d(x,y,p) : \Theta(x,y,p)}$$

$$\frac{x,y:\Phi,\ p:\mathrm{Id}_\Phi(x,y) \vdash \Theta(x,y,p):\mathsf{Cxt} \qquad x:\Phi \vdash d(x) : \Theta(x,x,r_\Phi(x))}{x:\Phi \vdash J_d(x,x,r_\Phi(x)) = d(x) : \Theta(x,x,r_\Phi(x))}$$

We start by taking $\mathrm{Id}_\Phi(a,b) := \mathrm{Id}_A(a,b)$ (for $a,b:A$) and $r_\Phi(a) := r_A(a)$. In this case, Id Cxt−ELIM corresponds to the judgment

$$(9) \qquad \frac{x,y:A,\ p:\mathrm{Id}_A(x,y) \vdash \Theta(x,y,p):\mathsf{Cxt} \qquad x:A \vdash d(x) : \Theta(x,x,r_A(x))}{x,y:A,\ p:\mathrm{Id}_A(x,y) \vdash J_d(x,y,p) : \Theta(x,y,p)},$$

while Id Cxt−COMP asserts that, for $a : A$, $J_d(a,a,r_A(a)) = d(a)$. To derive (9) we proceed by induction on the length m of Θ. Again, there is nothing to say when $\Theta = [\,]$ and, if $m = 1$, the elimination rule (9) and the corresponding Id Cxt−COMP are a particular case of Id −ELIM and Id −COMP in (4) respectively.[1] Suppose then that we could derive (9) and the associated computation rule for contexts Θ of length $m \geq 1$ and take a context $x,y : A,\ p : \mathrm{Id}_A(x,y) \vdash \Theta(x,y,p):\mathsf{Cxt}$ of length $m+1$. This means that we can write

$$\Theta(x,y,p) = [u:\Lambda(x,y,p),\ v:D(x,y,p,u)],$$

where Λ is a context of length m and D is a type depending upon $x,y : A,\ p : \mathrm{Id}_A(x,y)$ and $u:\Lambda(x,y,p)$. The premise

$$x:A \vdash d(x) : \Theta(x,x,r_\Phi(x))$$

then translates into

$$x:A \vdash d_\Lambda(x):\Lambda(x,x,r_A(x)) \quad \text{and} \quad x:A \vdash d_D(x):D(x,x,r_A(x),d_\Lambda(x)).$$

Thus, by induction on Λ, we can derive the judgment

$$\frac{x,y:A,\ p:\mathrm{Id}_A(x,y) \vdash \Lambda(x,y,p):\mathsf{Cxt} \qquad x:A \vdash d_\Lambda(x) : \Lambda(x,x,r_A(x))}{x,y:A,\ p:\mathrm{Id}_A(x,y) \vdash J_{d_\Lambda}(x,y,p) : \Lambda(x,y,p)}$$

so that $J_{d_\Lambda}(x,x,r_A(x)) = d_\Lambda(x)$. Thus, by using Id −ELIM as in (4), we get

$$\frac{\begin{array}{c} x,y:A,\ p:\mathrm{Id}_A(x,y) \vdash D(x,y,p,J_{d_\Lambda}(x,y,p)):\mathsf{Type} \\ x:A \vdash d_D(x) : D(x,x,r_A(x),d_\Lambda(x)) = D(x,x,r_A(x),J_{d_\Lambda}(x,x,r_A(x)) \end{array}}{x,y:A,\ p:\mathrm{Id}_A(x,y) \vdash J_{d_D}(x,y,p) : D(x,y,p,J_{d_\Lambda}(x,y,p))}$$

with $J_{d_D}(x,x,r_A(x)) = d_D(x)$.[2] Setting $J_d(x,y,p) := (J_{d_\Lambda}(x,y,p),\ J_{d_D}(x,x,r_A(x))):\Theta(x,y,p)$, we conclude that (9) holds. The associated computation rule ($J_d(a,a,r_A(a)) = d(a)$, for $a:A$) follows from the computation rules for $J_{d_\Lambda}(x,y,p)$ and $J_{d_D}(x,y,p)$. This concludes the proof of (8) when the context Φ has length 1.

We can now proceed with the induction step on the length of the context Φ appearing in (7) as follows. Assume that (7) holds true for contexts of length n and pick a context Φ of length $n+1$. We will show that Φ verifies (8) (recall the comment at the beginning of the proof about our simplification to the case where the additional context Δ in (7) is empty). We can write $\Phi = [x_1 : \Lambda, x_2 : D(x_1)]$, where Λ is a context of length n and D is a type in context Λ. Now, the inductive hypothesis on Λ verifying (8) (where Φ is substituted with Λ) ensures that the proof we gave of Proposition 1.7 applies verbatim[3] to show what follows. Given a dependent context

[1] Note that this step would be true also if we had not chosen to set $\Delta = [\,]$ to start with.

[2] Again, observe that this step would have been true also if we had not chosen to set $\Delta = [\,]$ to start with.

[3] As long as all occurrences of "Type" must be substituted with "Cxt".

$x\!:\!\Lambda \vdash \Omega(x)\!:\!\mathsf{Cxt}$, we can derive the following judgment

$$x, y\!:\!\Lambda,\ p\!:\!\mathrm{Id}_\Lambda(x,y),\ z\!:\!\Omega(y) \vdash p^*(z)\!:\!\Omega(x)$$

with associated computation rule $(r_\Lambda(x))^*(z) = z : \Omega(x)$. In order to define Id_Φ it is enough to provide a judgment

$$x_1\!:\!\Lambda,\ y_1\!:\!D(x_1),\ x_2\!:\!\Lambda,\ y_2\!:\!D(x_2) \vdash \mathrm{Id}_\Phi(x_1,y_1,x_2,y_2)\!:\!\mathsf{Cxt}$$

and we can accomplish this task by defining

$$\mathrm{Id}_\Phi(x_1,y_1,x_2,y_2) := \Big[p\!:\!\mathrm{Id}_\Lambda(x_1,x_2),\ q\!:\!\mathrm{Id}_{D(x_1)}(y_1, p^*(y_2))\Big]. \tag{10}$$

Keep on going, we obtain $r_\Phi(x,y)$ (for $(x,y)\!:\!\Phi$) via the judgments

$$x\!:\!\Lambda,\ y\!:\!D(x) \vdash r_\Lambda(x)\!:\!\mathrm{Id}_\Lambda(x,x),$$

$$x\!:\!\Lambda,\ y\!:\!D(x) \vdash r_{D(x)}(y)\!:\!\mathrm{Id}_{D(x)}(y,(r_\Lambda(x))^*(y)) = \mathrm{Id}_{D(x)}(y,y).$$

In order to get the simplified elimination and computation rule for $\mathrm{Id}\,\mathsf{Cxt}$ as in (8), we will use the fact that in the inductive hypothesis we are assuming that Λ verifies the full version (7) of the identity context rules. We make the judgment

$$x_1, x_2\!:\!\Lambda,\ p\!:\!\mathrm{Id}_\Lambda(x_1,x_2) \vdash \Delta(x_1,x_2,p)\!:\!\mathsf{Cxt},$$

where

$$\Delta(x_1,x_2,p) := [y_1\!:\!D(x_1),\ y_2\!:\!D(x_2),\ q\!:\!\mathrm{Id}_{D(x_1)}(y_1,p^*(y_2))].$$

We can use such a Δ to rewrite the premises of the elimination rule in (8) as

$$\begin{array}{c} x_1,x_2\!:\!\Lambda,\ p\!:\!\mathrm{Id}_\Lambda(x_1,x_2),\ z\!:\!\Delta(x_1,x_2,p) \vdash \Theta(x_1,x_2,p,z)\!:\!\mathsf{Cxt} \\ x\!:\!\Lambda,\ y\!:\!D(x) \vdash d(x,y)\!:\!\Theta(x,x,r_\Lambda(x),y,y,r_{D(x)}(y)) \end{array} \tag{11}$$

We now want to apply $\mathrm{Id}\,\mathsf{Cxt}-\mathrm{ELIM}$ for Λ with one of the premise being the first premise in (11) above. Thus, we must make a judgment

$$x\!:\!\Lambda,\ y_1,y_2\!:\!D(x),\ q\!:\!\mathrm{Id}_{D(x)}(y_1,y_2) \vdash d'(x,y_1,y_2,q)\!:\!\Theta(x,x,r_\Lambda(x),y_1,y_2,q)$$

for an appropriate d' (note that $y_1,y_2 : D(x),\ q:\mathrm{Id}_{D(x)}(y_1,y_2)$ is just $\Delta(x,x,r_\Lambda(x))$). We get such a d' through a specific instance of (9) for $D(x)$ (seen as a context $[y\!:\!D(x)]$ and when we are allowing the presence of an additional background context, namely Λ):

$$\frac{\begin{array}{c} x\!:\!\Lambda,\ y_1,y_2\!:\!D(x),\ q\!:\!\mathrm{Id}_{D(x)}(y_1,y_2) \vdash \Theta(x,x,r_\Lambda(x),y_1,y_2,q)\!:\!\mathsf{Cxt} \\ x\!:\!\Lambda,\ y\!:\!D(x) \vdash d(x,y)\!:\!\Theta(x,x,r_\Lambda(x),y,y,r_{D(x)}(y)) \end{array}}{x\!:\!\Lambda,\ y_1,y_2\!:\!D(x),\ q\!:\!\mathrm{Id}_{D(x)}(y_1,y_2) \vdash d'(x,y_1,y_2,q)\!:\!\Theta(x,x,r_\Lambda(x),y_1,y_2,q)}$$

with computation rule asserting that $d'(x,y,y,r_{D(x)}(y)) = d(x,y)$. By the induction hypothesis on Λ, we can now derive the judgment

$$\frac{\begin{array}{c} x_1,x_2\!:\!\Lambda,\ p\!:\!\mathrm{Id}_\Lambda(x_1,x_2),\ z\!:\!\Delta(x_1,x_2,p) \vdash \Theta(x_1,x_2,p,z)\!:\!\mathsf{Cxt} \\ x\!:\!\Lambda, z\!:\!\Delta(x,x,r_\Lambda(x)) \vdash d'(x,z)\!:\!\Theta(x,x,r_\Lambda(x),z) \end{array}}{x_1,x_2\!:\!\Lambda,\ p\!:\!\mathrm{Id}_\Lambda(x_1,x_2),\ z\!:\!\Delta(x_1,x_2,p) \vdash J_{d'}(x_1,x_2,p,z)\!:\!\Theta(x_1,x_2,p,z)}$$

which shows the validity of the elimination rule in (8) for Φ. Finally, from the computation rule for Id_Λ, we have $J_{d'}(x,x,r_\Lambda(x),z) = d'(x,z)$ and a substitution of z with $(y,y,r_{D(x)}(y))$ together with the computation rule for d' give the computation rule in (8) for Φ. The proof is then complete. □

For sake of reference, we record here a result which we noticed to be true during the proof of Theorem 1.8 above.

Lemma 1.9. *Suppose given judgments* $\Gamma \vdash \Lambda : \mathsf{Cxt}$ *and* $\Gamma, x : \Lambda \vdash \Omega(x) : \mathsf{Cxt}$ *in* $\mathbb{T}$*. Then the following judgments are derivable in* $\mathbb{T}$*.*

$$\frac{\Gamma \vdash a, b : \Lambda \qquad \Gamma \vdash p : \mathrm{Id}_\Lambda(a,b) \qquad \Gamma \vdash w : \Omega(b)}{\Gamma \vdash p^*(w) : \Omega(a)} *\mathsf{Cxt}\text{-ELIM} \tag{12}$$

$$\frac{\Gamma \vdash a : \Lambda \qquad \Gamma \vdash v : \Omega(a)}{\Gamma \vdash (r_\Lambda)^*(v) = v : \Omega(a)} *\mathsf{Cxt}\text{-COMP} \tag{13}$$

□

There is a variant of the above Lemma which can be interpreted by saying that we are allowed to transport (dependent) elements of (dependent) contexts along paths.

Lemma 1.10. [GG08, Lemma 5] *Suppose given judgments* $\Gamma \vdash \Phi : \mathsf{Cxt}$ *and* $\Gamma, x : \Omega \vdash \Omega(x) : \mathsf{Cxt}$ *in* $\mathbb{T}$ *and* $\Gamma,\ x : \Phi \vdash f(x) : \Psi(x)$ *(so that* Ψ *is in particular a dependent context relative to* $[\Gamma, \Phi]$*). Then the following judgments are derivable in* $\mathbb{T}$*.*

$$\frac{\Gamma \vdash a, b : \Phi \qquad \Gamma \vdash p : \mathrm{Id}_\Phi(a,b) \qquad \Gamma \vdash v : \Omega(a)}{\Gamma \vdash p_!(v) : \Omega(b)} \ !\text{-ELIM}$$
$$\frac{\Gamma \vdash a : \Phi \qquad \Gamma \vdash v : \Omega(a)}{\Gamma \vdash (r_\Phi)_!(v) = v : \Omega(a)} \ !\text{-COMP} \tag{14}$$

$$\frac{\Gamma,\ x : \Phi \vdash f(x) : \Psi(x)}{\Gamma,\ x, x' : \Phi,\ u : \mathrm{Id}_\Phi(x, x') \vdash \mathsf{dep\text{-}cong}(x.f(x); u) : \mathrm{Id}_{\Omega(x')}(u_!(f(x)),\ f(x'))} \tag{15}$$

Proof. Using weakening of contexts and the elimination rule in (7), we can infer

$$\frac{\Gamma,\ x, y : \Phi,\ u : \mathrm{Id}_\Phi(x,y),\ z : \Omega(x) \vdash \Omega(y) : \mathsf{Cxt} \qquad \Gamma,\ x : \Phi,\ z : \Omega(x) \vdash z : \Omega(x)}{\Gamma,\ x, y : \Phi,\ u : \mathrm{Id}_\Phi(x,y),\ z : \Omega(x) \vdash J_{x.z}(x, y, u) : \Omega(y)}$$

with $J_{x.z}(x/x, x/y, r_\Phi(x)/u) = z : \Omega(x)$. Setting $p_!(v) := J_{x.z}(a/x, b/y, p/u) : \Omega(b)$, we conclude that the rules in (14) hold. We get dep-cong via an instance of Id Cxt-ELIM with premises

$$\Gamma,\ x, x' : \Phi,\ u : \mathrm{Id}_\Phi(x, x') \vdash \mathrm{Id}_{\Omega(x')}(u_!(f(x)), f(x')) : \mathsf{Type}$$

and

$$\Gamma, x : \Phi \vdash r_{\Omega(x)}(f(x)) : \mathrm{Id}_{\Omega(x)}(f(x), f(x)).$$

□

Remark 1.11. One can use Lemma 1.10 to derive another form of the identity context for a context Φ. Indeed, in the inductive step of the proof for Theorem 1.8, given a context Λ of length $n \geq 1$ and a context $\Phi = [x : \Lambda,\ y : D(x)]$, if $(x_1, y_1),\ (x_2, y_2) : \Lambda$, we could have set

$$\mathrm{Id}'_\Phi(x_1, y_1, x_2, y_2) := \left[p : \mathrm{Id}_\Lambda(x_1, x_2),\ q : \mathrm{Id}_{D(x_1)}(p_!(y_1), y_2) \right].$$

with canonical term $r'_\Phi(x, y) = (r_\Lambda(x),\ r_{D(x)}(y))$. It is easy to see, by modifying in the obvious manner the definition of the context Δ in the proof of Theorem 1.8 and varying the proof accordingly, that Id Cxt−ELIM and Id Cxt−COMP hold true also for this identity context Id'_Φ. We may use this other version of the identity context later on.

We have thus accomplished our task of obtaining contextual versions of all the axioms (2), (3) and (4).

We end this section by giving the definition of mapping cylinders for contexts. These are examples of *higher inductive contexts* and will be crucial to prove that $\mathcal{C}\ell(\mathbb{T})$ admits a pre-model structure (see Appendix A).

Definition 1.12. [Lum11, cf. Definition 4] A dependent type theory $\mathbb{T}$ with identity types has *(non-dependent) mapping cylinders (for contexts)* if it has constructors for contexts Cyl, in-base, in-top and in-cyl satisfying the following rules (where Φ and Ψ are contexts):

$$\frac{x:\Phi \vdash f(x):\Psi}{y:\Psi \vdash \mathsf{Cyl}_{x.f(x)}(y):\mathsf{Cxt}}\ \mathsf{Cyl}\text{-}\textsc{Form}$$

$$\frac{x:\Phi \vdash f(x):\Psi}{y:\Psi \vdash \mathsf{in\text{-}base}(y):\mathsf{Cyl}_{x.f(x)}(y)}\ \mathsf{in\text{-}base}\text{-}\textsc{Intro} \qquad \frac{x:\Phi \vdash f(x):\Psi}{x:\Phi \vdash \mathsf{in\text{-}top}(x):\mathsf{Cyl}_{x.f(x)}(f(x))}\ \mathsf{in\text{-}top}\text{-}\textsc{Intro}$$

$$\frac{x:\Phi \vdash f(x):\Psi}{x:\Phi \vdash \mathsf{in\text{-}cyl}(x):\mathrm{Id}_{\mathsf{Cyl}_{x.f(x)}(f(x))}(\mathsf{in\text{-}top}(x),\ \mathsf{in\text{-}base}(f(x)))}\ \mathsf{in\text{-}cyl}\text{-}\textsc{Intro}$$

$$\frac{\begin{array}{c} x:\Phi \vdash f(x):\Psi \qquad y:\Psi,\ z:\mathsf{Cyl}_{x.f(x)}(y) \vdash \Theta(y,z):\mathsf{Cxt} \\ y:\Psi \vdash d_{\mathsf{base}}(y):\Theta(y,\mathsf{in\text{-}base}(y)) \qquad x:\Phi \vdash d_{\mathsf{top}}(x):\Theta(f(x),\ \mathsf{in\text{-}top}(x)) \\ x:\Phi \vdash d_{\mathsf{cyl}}(x):\mathrm{Id}_{\Theta(f(x)/y,\ \mathsf{in\text{-}base}(f(x))/z)}(\mathsf{in\text{-}cyl}(x)_!(d_{\mathsf{top}}(x)),\ d_{\mathsf{base}}(f(x))) \end{array}}{y:\Psi,\ z:\mathsf{Cyl}_{x.f(x)}(y) \vdash \mathsf{cyl\text{-}elim}_{\Theta}(d_{\mathsf{base}},d_{\mathsf{top}},d_{\mathsf{cyl}};y,z):\Theta(y,z)}\ \mathsf{Cyl}\text{-}\textsc{Elim}$$

$$\frac{\begin{array}{c} x:\Phi \vdash f(x):\Psi \qquad y:\Psi,\ z:\mathsf{Cyl}_{x.f(x)}(y) \vdash \Theta(y,z):\mathsf{Cxt} \\ y:\Psi \vdash d_{\mathsf{base}}(y):\Theta(y,\mathsf{in\text{-}base}(y)) \qquad x:\Phi \vdash d_{\mathsf{top}}(x):\Theta(f(x),\ \mathsf{in\text{-}top}(x)) \\ x:\Phi \vdash d_{\mathsf{cyl}}(x):\mathrm{Id}_{\Theta(f(x)/y,\ \mathsf{in\text{-}base}(f(x))/z)}(\mathsf{in\text{-}cyl}(x)_!(d_{\mathsf{top}}(x)),\ d_{\mathsf{base}}(f(x))) \end{array}}{y:\Psi \vdash \mathsf{cyl\text{-}elim}_{\Theta}(d_{\mathsf{base}},d_{\mathsf{top}},d_{\mathsf{cyl}};y,\mathsf{in\text{-}base}(y)):\Theta(y,\mathsf{in\text{-}base}(y))}\ \mathsf{Cyl}\text{-}\textsc{Comp1}$$

$$\frac{\begin{array}{c} x:\Phi \vdash f(x):\Psi \qquad y:\Psi,\ z:\mathsf{Cyl}_{x.f(x)}(y) \vdash \Theta(y,z):\mathsf{Cxt} \\ y:\Psi \vdash d_{\mathsf{base}}(y):\Theta(y,\mathsf{in\text{-}base}(y)) \qquad x:\Phi \vdash d_{\mathsf{top}}(x):\Theta(f(x),\ \mathsf{in\text{-}top}(x)) \\ x:\Phi \vdash d_{\mathsf{cyl}}(x):\mathrm{Id}_{\Theta(f(x)/y,\ \mathsf{in\text{-}base}(f(x))/z)}(\mathsf{in\text{-}cyl}(x)_!(d_{\mathsf{top}}(x)),\ d_{\mathsf{base}}(f(x))) \end{array}}{x:\Phi \vdash \mathsf{cyl\text{-}elim}_{\Theta}(d_{\mathsf{base}},d_{\mathsf{top}},d_{\mathsf{cyl}};f(x),\mathsf{in\text{-}top}(x)):\Theta(f(x),\mathsf{in\text{-}top}(x))}\ \mathsf{Cyl}\text{-}\textsc{Comp2}$$

$$\frac{\begin{array}{c} x:\Phi \vdash f(x):\Psi \qquad y:\Psi,\ z:\mathsf{Cyl}_{x.f(x)}(y) \vdash \Theta(y,z):\mathsf{Cxt} \\ y:\Psi \vdash d_{\mathsf{base}}(y):\Theta(y,\mathsf{in\text{-}base}(y)) \qquad x:\Phi \vdash d_{\mathsf{top}}(x):\Theta(f(x),\ \mathsf{in\text{-}top}(x)) \\ x:\Phi \vdash d_{\mathsf{cyl}}(x):\mathrm{Id}_{\Theta(f(x)/y,\ \mathsf{in\text{-}base}(f(x))/z)}(\mathsf{in\text{-}cyl}(x)_!(d_{\mathsf{top}}(x)),\ d_{\mathsf{base}}(f(x))) \end{array}}{x:\Phi \vdash \mathsf{cyl\text{-}comp}(z):\mathrm{Id}(\mathsf{dep\text{-}cong}(z.\mathsf{cyl\text{-}elim}_{\Theta}(d_{\mathsf{base}},d_{\mathsf{top}},d_{\mathsf{cyl}};f(x),z);\mathsf{in\text{-}cyl}(x)),\ d_{\mathsf{cyl}}(x))}\ \mathsf{Cyl}\text{-}\textsc{Comp3}$$

Remark 1.13. Note that the first and the second computation rules in Definition 1.12 above posit definitional equality, whereas the last computation rule only exhibits a path.

Notation 1.14. For sake of readability, we may omit in the following the subscript $x.f(x)$ from Cyl and just write $\mathsf{Cyl}_f(y)$ for $y:\Psi$.

2. The classifying category of a dependent type theory

Given a dependent type theory $\mathbb{T}$ (with identity types) as in Section 1 above, the next step is to define a category retaining the same syntactic information as $\mathbb{T}$. We first need the following

Definition 2.1. Let

$$\Phi = [x_0:A_0,\ \dots, x_n:A_n(x_0,\ \dots,\ x_{n-1})] \quad \text{and} \quad \Psi = [y_0:B_0,\ \dots, y_m:B_m(x_0,\ \dots,\ x_{m-1})]$$

be contexts. A *(context) morphism* $f\colon \Phi \to \Psi$ is a sequence

$$f = (f_0:B_0,\ \dots,\ f_m:B_m(f_0,\ \dots,\ f_{m-1}))$$

of terms such that

$$\Phi \vdash f_0:B_0, \quad \forall i \in \{1,\ \dots,\ m\}\ (\Phi \vdash f_i:B_i(f_0,\ \dots,\ f_{i-1}))\,.$$

Definition 2.2. The *classifying category* (or the *syntactic category*) associated to the dependent type theory $\mathbb{T}$ is the category $\mathcal{C}\ell(\mathbb{T})$ defined as follows.

(i) The objects of $\mathcal{C}\ell(\mathbb{T})$ are equivalences classes of contexts, where two contexts

$$\Phi = [x_0 : A_0, \ldots, x_n : A_n(x_0, \ldots, x_{n-1})]$$

and

$$\Psi = [y_0 : B_0, \ldots, y_m : B_m(y_0, \ldots, y_{m-1})]$$

are in the same equivalence class if they differ by the naming of their free variables[4] or if they are componentwise definitionally equal (see Definition 1.6). We will not distinguish notationally between a context and its equivalence class in $\mathcal{C}\ell(\mathbb{T})$.

(ii) A morphism from Φ to Ψ in $\mathcal{C}\ell(\mathbb{T})$ is an equivalence class of context morphisms $\Phi \to \Psi$, where two context morphisms

$$f = (f_0, \ldots, f_m),\ f = (f_0', \ldots, f_m') \colon \Phi \to \Psi$$

are in the same equivalence class if they differ by renaming of free variables in Φ or if they are pointwise definitionally equal, i.e. for all $i \in \{0, \ldots, m\}$,

$$\Phi \vdash f_i = f_i' : B_i(y_0, \ldots, y_{i-1}).$$

Again, we will not distinguish notationally between a context morphism and its equivalence class in $\mathcal{C}\ell(\mathbb{T})$.

(iii) The composition of $f \colon \Phi \to \Psi$ and $g \colon \Psi \to \Theta$ is given by substitution. Namely, if Φ and Ψ are as above, whereas

$$\Theta = [z_0 : C_0, \ldots, z_k : C_k(z_0, \ldots, z_{k-1})]$$

and we have $f = (f_0, \ldots, f_m)$ and $g = (g_0, \ldots, g_k)$, then $g \circ f$ is the context morphism

$$g \circ f = (g_0[f/\bar{y}], \ldots, g_k[f/\bar{y}]).$$

Here, with $\bar{y}$ we mean the free variables $[y_0, \ldots, y_m]$ appearing in Ψ and with $g_i[f/\bar{y}]$ (for $i \in \{0, \ldots, k\}$) we mean the result of substituting in g_i each occourence of y_j with f_j, for $j \in \{0, \ldots, m\}$. Such a composition is well defined on equivalence classes because substitution respects definitional equality, as it is inferable from the substitution rule in (2).

(iv) The identity in $\mathcal{C}\ell(\mathbb{T})$ at a context Φ is the context morphism $\mathrm{id}_\Phi = (x_0, \ldots, x_n) \colon \Phi \to \Phi$, where $\Phi = [x_0 : A_0, \ldots, x_n : A_n(x_0, \ldots, x_{n-1})]$ as above.

Remark 2.3. It follows immediately from the definition of a context morphism that the *empty context* $[\,]$ is the terminal object in the classifying category $\mathcal{C}\ell(\mathbb{T})$.

A fundamental role throughout our work will be played by the following special kind of context morphism

Definition 2.4. Given a context $\Gamma = [x_0 : A_0, \ldots, x_n : A_n(x_0, \ldots, x_{n-1})]$ and a judgment $\Gamma \vdash x : A$, the *display map* (*associated to* $\Gamma \vdash x : A$) is the context morphism

$$\pi_A = (x_0, \ldots, x_n) \colon [\Gamma,\ x : A] \to \Gamma,$$

where $[\Gamma,\ x : A]$ is a shortcut for the context $[x_0 : A_0, \ldots, x_n : A_n(x_0, \ldots, x_{n-1}),\ x : A]$. (In writing $[x_0 : A_0, \ldots, x_n : A_n(x_0, \ldots, x_{n-1}),\ x : A]$ we are implicitely assuming that x is not one of the variables $x_0, \ldots, x_n$ already appearing in Γ).

[4] The process of renaming free variables is also known as *α-conversion*.

We will commonly abuse of language and just speak of a *display map* as any map of the form $\pi_A\colon [\Gamma,\ x\colon A] \to \Gamma$ as the one above, thus tacitly assuming that the judgment $\Gamma \vdash x\colon A$ is derivable.

We will also need a generalization of the above notion.

Definition 2.5. Let Γ be a context and let $\Gamma \vdash \Phi\colon \mathsf{Cxt}$ (see Definition 1.4). The *dependent projection (associated to)* Φ is the context morphism

$$\pi_\Phi\colon [\Gamma,\ \Phi] \to \Gamma$$

consisting of the variables of Γ seen as terms.

Also in this case, we will commonly abuse of language and just talk about *dependent projections* as those maps of the form $\pi_\Phi\colon [\Gamma,\ \Phi] \to \Gamma$ as the one above, without stating explicitly that Φ is a dependent context relative to Γ.

Remark 2.6. By definition, it is clear that a dependent projection is a composite of display maps. In other words, if we close the class of display maps under composition we get precisely the class of dependent projections.

Remark 2.7. Observe that a dependent term of a dependent context Φ relative to Γ (see Definition 1.5) is a section of the dependent projection $\pi_\Gamma\colon [\Gamma,\ \Phi] \to \Gamma$, i.e. it is an element of

$$(\mathcal{C}\ell(\mathbb{T})/\Gamma)\,(1_\Gamma,\ \pi_\Gamma)\,,$$

where $\mathcal{C}\ell(\mathbb{T})/\Gamma$ is the slice category of $\mathcal{C}\ell(\mathbb{T})$ over Γ. Note also that a dependent term of $[x : A]$ relative to Γ is just a term a of type A.

Here is the first reason for which we care about display maps and dependent projections.

Proposition 2.8. [Pit00, Lemma 6.3.2] *Pullbacks along display maps exist in $\mathcal{C}\ell(\mathbb{T})$. More precisely, given any display map $\pi_A\colon [\Gamma,\ x : A] \to \Gamma$ and any context morphism $f\colon \Delta \to \Gamma$, the commutative diagram*

$$\begin{array}{ccc}
[\Delta,\ y\colon A[f/\bar{x}]] & \xrightarrow{(f,\,y)} & [\Gamma,\ x\colon A] \\
{\scriptstyle \pi_{A[f/\bar{x}]}}\big\downarrow & & \big\downarrow{\scriptstyle \pi_A} \\
\Delta & \xrightarrow[f]{} & \Gamma
\end{array}$$

is a pullback square in $\mathcal{C}\ell(\mathbb{T})$. (In the above diagram, we wrote $\bar{x}$ to mean the list of free variables appearing in Γ).

Proof. The commutativity of the diagram is clear. Given morphisms $g\colon \Phi \to [\Gamma,\ x\colon A]$ and $h\colon \Phi \to \Delta$ with $\pi_A \circ g = f \circ h$, this same equality means that g is of the form $(f \circ h,\ g_A)$ for some term g_A such that $\Phi \vdash g_A\colon A[(f\circ h)/\bar{x}]$. Since $\Phi \vdash A[(f \circ h)/\bar{x}] = (A[f/\bar{x}])\,[h/\bar{x}']\colon \mathsf{Type}$, where $\bar{x}'$ is the list of free variables in Φ, we get the desired unique factorization

$$\langle h, g\rangle = (h,\ g_A)\colon \Phi \to [\Delta,\ y\colon A[f/\bar{x}]].$$

□

Corollary 2.9. *(1) Pullbacks along dependent projections exist in $\mathcal{C}\ell(\mathbb{T})$.*
(2) $\mathcal{C}\ell(\mathbb{T})$ has finite products.

Proof. The first part follows from Remark 2.6 and the pasting lemma for pullbacks. As for the second claim, we can construct binary products in $\mathcal{C}\ell(\mathbb{T})$ using pullbacks, since $\mathcal{C}\ell(\mathbb{T})$ has a terminal objects [ ] and, for each context Γ, the unique map $\Gamma \to [\ ]$ is a dependent projection. Thus, $\mathcal{C}\ell(\mathbb{T})$ has finite products. □

3. THE FUNDAMENTAL GROUPOID OF A CONTEXT

The identity context can be used to functorially equip any context with an attached 1−groupoid. This is done in essentially the same way which one uses to associate to every space (topological space or simplicial set) its fundamental groupoid. To stress this analogy and our intuition of contexts as spaces[5] (which lies at the basis of a homotopical interpretation of type theory), we give the following

Definition 3.1. Let Φ be a context and suppose given $a, b : \Phi$.

(1) A *path* from a to b (in Φ) is a term p of the context $\mathrm{Id}_\Phi(a,b)$. The terms a and b are also called the *endpoints* of the path p.
(2) Given $p, q : \mathrm{Id}_\Phi(a,b)$, a *homotopy* from p to q (in Φ) is a term θ of the context $\mathrm{Id}_{\mathrm{Id}_\Phi(a,b)}(p,q)$.
(3) Two paths p, q from a to b are said to be *homotopic* if the context $\mathrm{Id}_{\mathrm{Id}_\Phi(a,b)}(p,q)$ is inhabitated, i.e. if there is a term of that context.

Notation 3.2. In order to improve readability, from time to time we may avoid writing explicitly the contexts with respect to which the identities contexts are taken. For example, if $a, b : \Phi$, we may just write $\mathrm{Id}(a,b)$ to mean $\mathrm{Id}_\Phi(a,b)$.

Lemma 3.3. [GG08, Lemma 6] *Let Φ be a context in $\mathbb{T}$. Then the following judgments are inferable in $\mathbb{T}$.*

$$\text{(16)}\qquad \frac{\vdash a,b,c:\Phi \qquad \vdash p:\mathrm{Id}_\Phi(a,b) \qquad \vdash q:\mathrm{Id}_\Phi(b,c)}{\vdash q\circ p:\mathrm{Id}_\Phi(a,c)}\ \textsc{Path-Comp} \qquad \frac{\vdash a,b:\Phi \qquad \vdash p:\mathrm{Id}_\Phi(a,b)}{\vdash 1_b\circ p = p:\mathrm{Id}_\Phi(a,b)}$$

$$\text{(17)}\qquad \frac{\vdash a:\Phi}{1_a:\mathrm{Id}_\Phi(a,a)}\ \textsc{Path-Unit}$$

$$\text{(18)}\qquad \frac{\vdash a,b:\Phi \qquad \vdash p:\mathrm{Id}_\Phi(a,b)}{\vdash p^{-1}:\mathrm{Id}_\Phi(b,a)}\ \textsc{Path-Inv} \qquad \frac{\vdash a:\Phi}{\vdash (1_a)^{-1} = 1_a:\mathrm{Id}_\Phi(a,a)}$$

Proof. Since, for a fixed $a:\Phi$, $x:\Phi \vdash \mathrm{Id}(a,x):\mathsf{Cxt}$, in order to get (16) we can use (14) to derive

$$\frac{\vdash b,c:\Phi \qquad \vdash q:\mathrm{Id}_\Phi(b,c) \qquad \vdash p:\mathrm{Id}_\Phi(a,b)}{\vdash q_!(p):\mathrm{Id}_\Phi(a,c)}$$

with $(r_\Phi(b))_!(p) = p : \mathrm{Id}_\Phi(a,b)$, so that we can set $q\circ p := q_!(p) : \mathrm{Id}_\Phi(a,c)$. The rule (17) is just Id Cxt−INTRO if we set $1_a := r_\Phi(a)$. Finally, we obtain (18) using Id Cxt−ELIM to get the judgment

$$\frac{x,y:\Phi,\ u:\mathrm{Id}_\Phi(x,y) \vdash \mathrm{Id}_\Phi(y,x):\mathsf{Cxt} \qquad x:\Phi\vdash r_\Phi(x):\mathrm{Id}_\Phi(x,x)}{x,y:\Phi,\ u:\mathrm{Id}_\Phi(x,y)\vdash J_{x.r_\Phi(x)}(x,y,u) : \mathrm{Id}_\Phi(y,x)}$$

with $J_{x.r_\Phi(x)}(x/x, x/y, r_\Phi(x)/u) = r_\Phi(x):\mathrm{Id}_\Phi(x,x)$, so that we can take $p^{-1} := J_{x.r_\Phi(x)}(a/x, b/y, p/u)$. □

Definition 3.4. In the situation of Lemma 3.3 above, we will refer to the terms $q\circ p$, 1_a and p^{-1} as to the *composite path* of p and q, the *constant path* at a and the *inverse path* of p. (Note that $1_a := r_\Phi(a)$, by the proof of the above Lemma).

[5] A clarification is probably in order at this point: when we think to contexts as spaces, such an interpretation should rely only on *homotopical properties* of spaces and not purely topological ones (open or closed sets, separation properties and so on).

Remark 3.5. For simplicity, we formulated Lemma 3.3 without any ambient context Γ. However, as it is transparent from the proof we gave, (16), (17) and (18) still hold in presence of an additional context Γ on the left of each turnstile in the premises and in the conclusions of all the judgments. The same remark will apply to all the other rules concerning paths and homotopies that we will derive below.

Remark 3.6. Lemma 3.3 holds for arbitrary contexts so, in particular, it can be applied to $\mathrm{Id}_\Phi(a,b)$ (given the judgments $\vdash \Phi : \mathsf{Cxt}$ and $\vdash a,b:\Phi$) to deduce that we can (vertically) compose homotopies and that there are inverse homotopies and constant homotopies at paths. We get similar statements for homotopies between homotopies and then for the homotopies between those and so on. The most striking aspect of such a construction is that it provides all of this information in a purely syntactic way, by simply deducing it from the inference rules that constitute the background setting of our type theory.[6]

Next, we take a look at horizontal compositions and inverses of homotopies.

Lemma 3.7. [GG08, Lemma 7] *Let Φ be a context in $\mathbb{T}$ and suppose we are given judgments $\vdash a,b,c:\Phi$, $\vdash p_0,p_1:\mathrm{Id}_\Phi(a,b)$ and $\vdash q_0,q_1:\mathrm{Id}_\Phi(b,c)$. Then the following judgments are derivable in $\mathbb{T}$:*

$$\frac{\vdash \phi:\mathrm{Id}(p_0,p_1) \qquad \vdash \psi:\mathrm{Id}(q_0,q_1)}{\vdash \psi\bullet\phi:\mathrm{Id}(q_0\circ p_0, q_1\circ p_1)} \qquad\qquad \frac{\vdash \phi:\mathrm{Id}(p_0,p_1)}{\vdash \underline{\phi}:\mathrm{Id}(p_0^{-1},p_1^{-1})}$$

with associated computation rules

$$\frac{\vdash p:\mathrm{Id}(a,b) \qquad \vdash q:\mathrm{Id}(b,c)}{\vdash 1_q\bullet 1_p = 1_{q\circ p}:\mathrm{Id}(q\circ p, q\circ p)} \qquad\qquad \frac{\vdash p:\mathrm{Id}(a,b)}{\underline{1_p} = 1_{p^{-1}}:\mathrm{Id}(p^{-1},p^{-1})}$$

Proof. In order to get the first rule, we start by using $\mathrm{Id}\,\mathsf{Cxt}{-}\mathrm{ELIM}$ in (7) with Φ replaced by $\mathrm{Id}_\Phi(b,c)$ and auxiliary context Δ given by $p:\mathrm{Id}_\Phi(a,b)$, so that we can derive

$$\frac{\begin{array}{c} e:\mathrm{Id}(b,c),\ f:\mathrm{Id}(b,c),\ \sigma:\mathrm{Id}(e,f),\ p:\mathrm{Id}(a,b)\vdash \mathrm{Id}(e\circ p, f\circ p):\mathsf{Cxt} \\ e:\mathrm{Id}(b,c),\ p:\mathrm{Id}(a,b)\vdash 1_{e\circ p}:\mathrm{Id}(e\circ p, e\circ p)\end{array}}{e:\mathrm{Id}(b,c),\ f:\mathrm{Id}(b,c),\ \sigma:\mathrm{Id}(e,f),\ p:\mathrm{Id}(a,b)\vdash J_{e.1_{e\circ p}}(e,f,\sigma,p):\mathrm{Id}(e\circ p,e\circ p)}$$

with computation rule $J_{e.1_{e\circ p}}(e/e,e/f,1_e/\sigma,p) = 1_{e\circ p}$ (recall that $1_e = r_{\mathrm{Id}(b,c)}(e)$). We set, for $\sigma:\mathrm{Id}(e,f)$ and $p:\mathrm{Id}(a,b)$, $\sigma\cdot p := J_{e.1_{e\circ p}}(e,f,\sigma,p):\mathrm{Id}(e\circ p, f\circ p)$, so that $1_e\cdot p = 1_{e\circ p}$. Similarly, we can derive

$$\frac{\begin{array}{c} k:\mathrm{Id}(a,b),\ l:\mathrm{Id}(a,b),\ \tau:\mathrm{Id}(k,l),\ q:\mathrm{Id}(b,c)\vdash \mathrm{Id}(q\circ k, q\circ l):\mathsf{Cxt} \\ k:\mathrm{Id}(a,b),\ q:\mathrm{Id}(b,c)\vdash 1_{q\circ k}:\mathrm{Id}(q\circ k, q\circ k)\end{array}}{k:\mathrm{Id}(a,b),\ l:\mathrm{Id}(a,b),\ \tau:\mathrm{Id}(k,l),\ q:\mathrm{Id}(b,c)\vdash J_{k.1_{q\circ k}}(k,l,\tau,q):\mathrm{Id}(q\circ k,q\circ k)}$$

so that, if we set, for $\tau:\mathrm{Id}(k,l)$ and $q:\mathrm{Id}(a,b)$, $q\cdot\tau := J_{k.1_{q\circ k}}(k,l,\sigma,p):\mathrm{Id}(q\circ k, q\circ l)$, we get $q\cdot 1_k = 1_{q\circ k}$. We can now define

$$\psi\bullet\phi := (\psi\cdot p_0)\circ(q_1\cdot\phi).$$

which indeed verifies $1_q\bullet 1_p = 1_{q\circ p}\circ 1_{q\circ p} = 1_{q\circ p}$.
As for the second rule, we can infer

$$\frac{k:\mathrm{Id}(a,b),\ l:\mathrm{Id}(a,b),\ \tau:\mathrm{Id}(k,l)\vdash \mathrm{Id}(k^{-1},l^{-1}):\mathsf{Cxt} \qquad k:\mathrm{Id}(a,b)\vdash 1_{k^{-1}}:\mathrm{Id}(k^{-1},k^{-1})}{k:\mathrm{Id}(a,b),\ l:\mathrm{Id}(a,b),\ \tau:\mathrm{Id}(k,l)\vdash J_{k.1_{k^{-1}}}(k,l,\tau):\mathrm{Id}(k^{-1},l^{-1})}$$

so that, setting $\underline{\phi} := J_{k.1_{k^{-1}}}(p_0/k, p_1/l, \phi/\tau):\mathrm{Id}(p_0^{-1},p_1^{-1})$, we can complete our proof. □

Finally, we address associativity, unitality and existence of inverses up to homotopy of path composition.

[6] Seriously, this is marvelous!

Lemma 3.8. [GG08, Lemma 8] *Let Φ be a context in $\mathbb{T}$ and suppose given judgments $\vdash a,b,c,d:\Phi$. Then the following rules are inferable in $\mathbb{T}$*

$$\frac{\vdash p:\mathrm{Id}(a,b) \qquad \vdash q:\mathrm{Id}(b,c) \qquad \vdash r:\mathrm{Id}(c,d)}{\vdash \alpha_{p,q,r}:\mathrm{Id}((r\circ q)\circ p, r\circ(q\circ p))}$$

$$\frac{\vdash p:\mathrm{Id}(a,b)}{\vdash \phi_p:\mathrm{Id}(1_b\circ p, p)} \qquad \frac{\vdash p:\mathrm{Id}(a,b)}{\vdash \psi_p:\mathrm{Id}(p\circ 1_a, p)}$$

$$\frac{\vdash p:\mathrm{Id}(a,b)}{\vdash \sigma_p:\mathrm{Id}(p^{-1}\circ p, 1_a)} \qquad \frac{\vdash p:\mathrm{Id}(a,b)}{\vdash \tau_p:\mathrm{Id}(p\circ p^{-1}, 1_b)}$$

and they satisfy associated computation rules

$$\frac{\vdash p:\mathrm{Id}(a,b) \qquad \vdash q:\mathrm{Id}(b,c)}{\vdash \alpha_{p,q,1_c} = 1_{q\circ p}:\mathrm{Id}(q\circ p, q\circ p)}$$

$$\frac{\vdash a:\Phi}{\phi_{1_a} = 1_{1_a}:\mathrm{Id}(1_a,1_a)} \qquad \frac{\vdash a:\Phi}{\psi_{1_a} = 1_{1_a}:\mathrm{Id}(1_a,1_a)}$$

$$\frac{\vdash a:\Phi}{\sigma_{1_a} = 1_{1_a}:\mathrm{Id}(1_a,1_a)} \qquad \frac{\vdash a:\Phi}{\tau_{1_a} = 1_{1_a}:\mathrm{Id}(1_a,1_a)}$$

Proof. As usual, we apply a suitable instance of Id Cxt−ELIM to derive

$$\frac{\begin{array}{c} x,y:\Phi,\ u:\mathrm{Id}(x,y),\ z,w:\Phi,\ t:\mathrm{Id}(z,w),\ s:\mathrm{Id}(w,x)\vdash \mathrm{Id}((u\circ s)\circ t, u\circ(s\circ t)):\mathsf{Cxt} \\ x:\Phi,\ z,w:\Phi,\ t:\mathrm{Id}(z,w),\ s:\mathrm{Id}(w,x)\vdash 1_{s\circ t}:\mathrm{Id}(s\circ t, s\circ t)\end{array}}{x,y:\Phi,\ u:\mathrm{Id}(x,y),\ z,w:\Phi,\ t:\mathrm{Id}(z,w),\ s:\mathrm{Id}(w,x)\vdash J_{x.1_{s\circ t}}(x,y,u,z,w,t,s):\mathrm{Id}((u\circ s)\circ t, u\circ(s\circ t))}$$

and we can set $\alpha_{p,q,r} := J_{x.1_{s\circ t}}(c/x,d/y,r/u,a/z,b/w,p/t,q/s) : \mathrm{Id}((r\circ q)\circ p, r\circ(q\circ p))$. The computation rule for α follows from the one for $J_{x.1_{s\circ t}}$. By Lemma 3.3 above, since $1_b\circ p = p$, we can take ϕ_p to just be 1_p. The term $\psi_p:\mathrm{Id}(p\circ 1_a, p)$ can be obtained via a simple application of the elimination rule for identity context with premises $x,y:\Phi,\ u:\mathrm{Id}(x,y)\vdash \mathrm{Id}(u\circ 1_x, u):\mathsf{Cxt}$ and $x:\Phi\vdash 1_{1_x}:\mathrm{Id}(1_x,1_x)$. Similarly, since, for $x:\Phi$, $1_x^{-1}\circ 1_x = 1_x\circ 1_x = 1_x$, we can get σ_p via Id Cxt−ELIM on the premises $x,y:\Phi,\ u:\mathrm{Id}(x,y)\vdash \mathrm{Id}(u^{-1}\circ u, 1_x)$ and $x:\Phi\vdash 1_{1_x}:\mathrm{Id}(1_x,1_x)$. The derivation of τ_p is totally analogous. □

Remark 3.9. Using a suitable variation of Id Cxt−ELIM, given p,q and r as in Lemma 3.8, we can also derive a term $\alpha'_{p,q,r}:\mathrm{Id}((r\circ q)\circ p, r\circ(q\circ p))$ with computing rules $\alpha'_{1_b,q,r}:\mathrm{Id}(r\circ q, r\circ q)$.

We are now ready to give the following

Definition 3.10. Let Φ be a context in $\mathbb{T}$. The *fundamental groupoid* associated to Φ is the groupoid $\Pi_1(\Phi)$ defined as follows.

(1) Objects of $\Pi_1(\Phi)$ are given by the terms $a:\Phi$.
(2) Morphisms between $a:\Phi$ and $b:\Phi$ in $\Pi_1(\Phi)$ are given by the equivalence classes $[p]$ of paths $p:\mathrm{Id}(a,b)$, where two paths $p,p':\mathrm{Id}(a,b)$ are equivalent if they are homotopic.
(3) Given $a,b,c:\Phi$, $p:\mathrm{Id}(a,b)$ and $q:\mathrm{Id}(b,c)$,

$$[q]\circ[p] := [q\circ p], \quad 1_a := [1_a], \quad [p]^{-1} := [p^{-1}].$$

Note that:

(i) By Lemma (3.3) applied to the context $\mathrm{Id}(a,b)$, "being homotopic" is indeed an equivalence relation between paths in Φ with the same endpoints a and b.
(ii) The fact that the composition, identity and inverse rule in $\Pi_1(\Phi)$ are well defined and give indeed a groupoid structure is the content of Lemma 3.7 and of Lemma 3.8.

We now want to show that the above construction can be made functorially. In order to do this, we need the following

Lemma 3.11. *Let $f\colon \Phi \to \Psi$ be a morphisms of contexts in $\mathbb{T}$ and suppose given $a, b : \Phi$. Then we can derive the following rules in $\mathbb{T}$*

$$\frac{\vdash p : \mathrm{Id}(a,b)}{\vdash f(p) : \mathrm{Id}(fa, fb)}$$

$$\frac{\vdash p : \mathrm{Id}(a,b) \qquad \vdash p' : \mathrm{Id}(a,b) \qquad \vdash \alpha : \mathrm{Id}(p,p')}{\vdash f(\alpha) : \mathrm{Id}(f(p), f(p'))}$$

satisfying the computation rules

$$\frac{\vdash a : \Phi}{\vdash f(1_a) = 1_{fa} : \mathrm{Id}(fa, fa)}$$

$$\frac{\vdash a : \Phi}{\vdash f(1_{1_a}) = 1_{1_{fa}} : \mathrm{Id}(1_{fa}, 1_{fa})}$$

Proof. The first rule follows from a single application of Id Cxt$-$ELIM with premises $x, y : \Phi,\ u : \mathrm{Id}(x,y) \vdash \mathrm{Id}(fx, fy) : \mathsf{Cxt}$ and $x : \Phi \vdash 1_{fx} : \mathrm{Id}(fx, fx)$. The second rule requires a double application of Id Cxt$-$ELIM. First, we can derive

$$\frac{x, y : \Phi,\ u : \mathrm{Id}(x,y) \vdash \mathrm{Id}(1_{fx}, f(u)) \qquad x : \Phi \vdash 1_{1_{fx}} : \mathrm{Id}(1_{fx}, 1_{fx})}{x, y : \Phi,\ u : \mathrm{Id}(x,y) \vdash J_{x.1_{1_{fx}}}(x, y, u) : \mathrm{Id}(1_{fx}, f(u))}$$

with $J_{x.1_{1_{fx}}}(x, x/y, 1_x/u) = 1_{1_{fx}}$. Then we can obtain the judgment

$$\frac{\begin{array}{c} x, y : \Phi,\ u : \mathrm{Id}(x,y),\ u' : \mathrm{Id}(x,y),\ H : \mathrm{Id}(u,u') \vdash \mathrm{Id}(f(u), f(u')) : \mathsf{Cxt} \\ x : \Phi,\ v : \mathrm{Id}(x,x),\ K : \mathrm{Id}(1_x, v) \vdash J_{x.1_{1_{fx}}}(x, x/y, u/v) : \mathrm{Id}(1_{fx}, f(v)) \end{array}}{x, y : \Phi,\ u : \mathrm{Id}(x,y),\ u' : \mathrm{Id}(x,y),\ H : \mathrm{Id}(u,u') \vdash J_{x.\ J_{x.1_{1_{fx}}}(x,x/y,u/v)}(x, y, u, u', H) : \mathrm{Id}(f(u), f(u'))}$$

Setting $f(\alpha) := J_{x.\ J_{x.1_{1_{fx}}}(x,x/y,u/v)}(a/x, b/y, p/u, p'/u', \alpha/H) : \mathrm{Id}(f(p), f(p'))$, we can conclude our proof. ∎

Proposition 3.12. [GG08, Proposition 9] *Given any dependent type theory $\mathbb{T}$ with axioms for identity types there is a fundamental-groupoid functor*

$$\Pi_1 \colon \mathcal{C}\ell(\mathbb{T}) \longrightarrow \mathsf{Gpd}, \quad \Phi \mapsto \Pi_1(\Phi),$$

where Gpd *is the category of small groupoids and functors between them.*

Proof. Given morphism $f\colon \Phi \to \Psi$ of contexts, we define $\Pi_1(f)\colon \Pi_1(\Phi) \to \Pi_1(\Psi)$ as follows:

- $(\Pi_1(f))(a) := fa \in \Pi_1(\Psi)$, for $a \in \Phi$;
- $(\Pi_1(f))([p]) := [f(p)]\colon fa \to fb$, for $[p]\colon a \to b$ in $\Pi_1(\Phi)$.

Note that the action of $\Pi_1(f)$ on morphisms in $\Pi_1(\Phi)$ is well defined by Lemma 3.11 above. In order to show that $\Pi_1(f)$ is a functor, we need to verify that, for $p:\mathrm{Id}(a,b)$ and for $q:\mathrm{Id}(b,c)$, we have

$$[f(1_a)] = [1_{fa}] \quad \text{and} \quad [f(q \circ p)] = [f(q) \circ f(p)].$$

The first equation holds before passing to homotopy classes by Lemma 3.11 again. As for the second one, it follows from Id Cxt−ELIM with premises $x, y : \Phi,\ u : \mathrm{Id}(x,y),\ z : \Phi,\ r : \mathrm{Id}(y,z) \vdash \mathrm{Id}(f(r \circ u), f(r) \circ f(u))$ and $x:\Phi,\ z:\Phi,\ r:\mathrm{Id}(x,z) \vdash 1_{f(r)} : \mathrm{Id}(f(r), f(r))$.
We now need to show that the assignments $\Phi \mapsto \Pi_1(\Phi)$ and $f \mapsto \Pi_1(f)$ do define a functor $\mathcal{C}\ell(\mathbb{T}) \longrightarrow \mathsf{Gpd}$. Given context morphisms $f\colon \Delta \to \Phi$ and $g\colon \Phi \to \Psi$, we must check that $\Pi_1(g \circ f) = \Pi_1(g) \circ \Pi_1(f)$. Certainly, $\Pi_1(g \circ f)$ and $\Pi_1(g) \circ \Pi_1(f)$ act in the same way on objects. To conclude that they are indeed the same functor it is then enough to show that, for $p:\mathrm{Id}(a,b)$, there is a homotopy between $g(f(p))$ and $(gf)(p)$. This can be done using Id Cxt−ELIM with premises $x, y : \Phi,\ u : \mathrm{Id}(x,y) \vdash \mathrm{Id}(g(f(u)), (gf)(u)) : \mathsf{Cxt}$ and $x : \Phi \vdash 1_{g(f(x))} : \mathrm{Id}(g(f(1_x)), (gf)(1_x))$, where the second judgment is well-typed because $g(f(1_x)) = g(1_{fx}) = 1_{g(fx)} = 1_{(gf)x} = (gf)(1_x)$. Finally, the fact that $\Pi_1(1_\Phi) = 1_{\Pi_1(\Phi)}$ boils down to show that there is a homotopy between $p:\mathrm{Id}(a,b)$ and $(1_\Phi)(p)$ which follows again by Id Cxt-ELIM, in a way that we hope the readers can figure out by themselves by now. □

Remark 3.13. In order to define $\Pi_1(\Phi)$ for a context Φ we had to quotient out paths by the homotopy relation (and we remark again that this is precisely what we do to construct the fundamental groupoid of a space). The reason for which we must consider paths up to homotopy is essentially because, for paths p, q and r with appropriate endpoints, we can not show that there is a definitional equality $(r \circ q) \circ p = r \circ (q \circ p)$, but only that there is a homotopy between the two sides of the alleged equality. Similarly, we can not verify that $f(q \circ p) = f(q) \circ f(p)$ (for a context morphism $f\colon \Phi \to \Psi$) on the nose, nor that Π_1 is indeed a functor without passing to homotopy classes of paths. This is also the main obstacle to the possibility of associating to each context Φ a *strict internal groupoid* to $\mathcal{C}\ell(\mathbb{T})$. More precisely, the problem is that, in general, identity types between paths (and therefore also identity contexts between paths) are not discrete types (we are working with a so-called *intensional type theory*), which means precisely that we can not reduce the notion of "being homotopic" to that of "being (definitionally) equal" for paths. However, if we force identity types between paths to be discrete, i.e. we force homotopic paths to be equal and we require that each homotopy in $\mathrm{Id}(p,p)$ for a path p is definitionally equal to the constant homotopy at p, then it is indeed possible to endow each context with the structure of an internal groupoid. This approach is explained in [Gar09, Section 3].

4. Some remarkable classes of maps

We are going to define here some specific classes of maps in $\mathcal{C}\ell(\mathbb{T})$ that we will need to construct the pre-model category structure on the classifying category. In order to define these maps, we do not need our type theory to have mapping cylinders for contexts (Definition 1.12), so we will stick here to our assumption that $\mathbb{T}$ is a dependent type theory with identity contexts. The presence of mapping cylinders will become essential only in the next section as they are required to provide one of the two weak factorization systems needed to get a pre-model category structure.

First of all, we define the maps that will become the weak equivalences in the pre-model structure for $\mathcal{C}\ell(\mathbb{T})$.

Definition 4.1. [Lum11, Definition 3] Suppose given a judgment $\Gamma \vdash \Delta, \Theta : \mathsf{Cxt}$. We will say that Δ is *equivalent* to Θ (or that Δ and Θ are *equivalent contexts*) if we can derive the following five judgments

$$\Gamma,\ x:\Delta \vdash f(x):\Theta$$

$$\Gamma,\ y:\Theta \vdash g(y):\Delta \qquad \Gamma,\ x:\Delta \vdash \eta_x : \mathrm{Id}_\Delta(g(fx), x)$$

$$\Gamma,\ y\!:\!\Theta \vdash g'(y)\!:\!\Delta \qquad \Gamma,\ y\!:\!\Theta \vdash \epsilon_y\!:\!\mathrm{Id}_\Theta(f(g'y),y).$$

We will abbreviate these judgments by writing

$$\Gamma \vdash (f,g,g',\eta,\epsilon)\ \Delta\ \mathsf{equiv}\ \Theta.$$

We will also write

$$\Gamma \vdash \mathsf{isEquiv}(f)$$

if the last four judgments above are derivable for some (g,g',η,ϵ). In this case, we will say that f is an *equivalence (of contexts)*, that g is a *left quasi-inverse* for f and that g' is a *right quasi-inverse* for f.

Remark 4.2. [Pro13, cf. Section 2.4] Note that the above definition of being an equivalence for a context morphism is somehow external, in the sense that it involves the use of an existential quantifier not encoded in the type theory itself. This is due to the fact that we are not allowing our dependent type theories to have any type constructor except for identity types. However, if $\mathbb{T}$ has Σ-types and Π-types, we can define $\mathsf{isEquiv}(f)$ internally as the type

$$\mathsf{isEquiv}(f) := \left(\sum_{(g\colon B\to A)}\prod_{(x\colon A)}\mathrm{Id}_A(g(fx),x)\right)\times\left(\sum_{(g'\colon B\to A)}\prod_{(y\colon B)}\mathrm{Id}_B(f(g'y),y)\right)$$

Here, $f\colon A\to B$ is a function between types and we can make this reduction by substituting contexts with the associated Σ-types (see Remark 1.2).

Remark 4.3. Suppose that $f\colon\Gamma\to\Delta$ is an equivalence of contexts as witnessed by (g,g',η,ϵ). Then every right quasi-inverse for f is also a left quasi-inverse. Indeed, suppose $h\colon\Delta\to\Gamma$ is a right quasi-inverse for f, as witnessed by $y:\Delta\vdash\epsilon'_{fx}:\mathrm{Id}_\Delta(f(hy),y)$. Then, given $x:\Gamma$, we have paths

$$(\eta_{h(fx)})^{-1}\!:\!\mathrm{Id}_\Gamma(hfx,gfhfx),\quad g(\epsilon'_y)\!:\!\mathrm{Id}_\Gamma(gfhfx,gfx),\quad \eta_x\!:\!\mathrm{Id}_\Gamma(gfx,x).$$

The composite of these paths gives us a path from hfx to x, showing that h is also a left quasi-inverse for f. In particular, we could have defined a context morphism $f\colon\Delta\to\Theta$ to be an equivalence if it admits a *two-sided inverse*, with the obvious meaning.

The next ingredient we need is the notion of a contractible context.

Definition 4.4. [Lum11, Definition 2] Suppose $\Gamma\vdash\Delta\!:\!\mathsf{Cxt}$ in $\mathbb{T}$. We write

$$\Gamma\vdash(a,\alpha)\ \mathsf{contract}\ \Delta$$

if the following judgments are derivable in $\mathbb{T}$:

$$x\!:\!\Gamma\vdash a(x)\!:\!\Delta,\qquad x\!:\!\Gamma,\ y,y'\!:\!\Delta\vdash\alpha(x,y,y')\!:\!\mathrm{Id}_\Delta(y,y').$$

We say that Δ is *contractible* (or that it is a *contractible context*) and we write $\Gamma\vdash\mathsf{isContr}(\Delta)$, if $\Gamma\vdash(a,\alpha)$ contract Δ for some pair $(a,\ \alpha)$ of (dependent) terms as above. The dependent path α is also called a *contraction* for Δ and the dependent term a is called a *canonical inhabitant*.

Remark 4.5. If $\mathbb{T}$ has product and Π-types, given a type A, the judgment $\mathsf{isContr}(A)$ can be described internally has the type

$$\mathsf{isContr}(A) := A\times\prod_{x,x'\colon A}\mathrm{Id}_A(x,x').$$

Remark 4.6. By considering a type as a context of length 1, Definition 4.4 above also provides a notion of contractibility for types.

Here is an easy equivalent characterization of contractible contexts.

Lemma 4.7. [Pro13, Lemma 3.11.3] *Given a context Δ (relative to another context Γ), the following are equivalent:*

(i) Δ *is contractible, in the sense of Definition 4.4;*
(ii) *there is a term* $y_0 : \Delta$ *such that, for each* $y : \Delta$, *there is a path from* y *to* y_0.

Proof. If Δ is contractible via (a, α), then $y_0 := a$ and the path $\alpha(y, y_0)$ shows that (ii) holds. Viceversa, if we have $y_0 : \Delta$ and a path $\beta_y : \mathrm{Id}_\Delta(y, y_0)$ for each $y : \Delta$, then a contraction for Δ is given by

$$y, y' : \Delta \vdash (\beta_{y'})^{-1} \circ \beta_y : \mathrm{Id}_\Delta(y, y')$$

with canonical inhabitant y_0. □

We can finally describe all the classes of maps that we will make use of. Given a class $\mathcal{A}$ of morphisms in $\mathcal{C}\ell(\mathbb{T})$ we use the notation ${}^{\pitchfork}\mathcal{A}$ (respectively $\mathcal{A}^{\pitchfork}$) to indicate the class of maps having the left lifting property (resp. having the right lifting property) with respect to all the maps in $\mathcal{A}$ (see Appendix A).

Definition 4.8. [Lum11, cf. Definition 11 - Definition 13][7] Let $\mathbb{T}$ be dependent type theory with identity types and let $\mathcal{F}_0$ be the class of dependent projections in $\mathcal{C}\ell(\mathbb{T})$ (see Definition 2.5) We define the the following classes of maps in $\mathcal{C}\ell(\mathbb{T})$:

(1) $\mathcal{W} := \{f : \Phi \to \Psi : \vdash \mathsf{isEquiv}(f)\}$.
(2) $\mathcal{TC} := {}^{\pitchfork}(\mathcal{F}_0)$.
(3) $\mathcal{F} = (\mathcal{TC})^{\pitchfork}$.
(4) $\mathcal{TF}_0 := \{\pi_\Delta : [\Gamma, \Delta] \to \Gamma \ : \ \pi_\Delta \text{ is a dependent projection and } \Gamma \vdash \mathsf{isContr}(\Delta)\}$.
(5) $\mathcal{C} = {}^{\pitchfork}(\mathcal{TF}_0)$.
(6) $\mathcal{TF} := \mathcal{C}^{\pitchfork}$.

We also define

(7) $\mathcal{F}'$ as the class of context morphisms $f : \Phi \to \Psi$ such that there is a dependent projection $\pi_\Phi : [\Gamma, \Phi] \to \Gamma$ for which f is isomorphic to π_Φ (in the arrow category of $\mathcal{C}\ell(\mathbb{T})$).

Remark 4.9. Since dependent projections are composite of display maps (see Remark 2.6), it follows that, if $\mathcal{D}$ is the class of display maps in $\mathcal{C}\ell(\mathbb{T})$, then $\mathcal{TC} = {}^{\pitchfork}\mathcal{D}$ and consequently $\mathcal{F} = \left({}^{\pitchfork}\mathcal{D}\right)^{\pitchfork}$.

Remark 4.10. Note the difference between $\mathcal{F}$ and $\mathcal{F}'$ in Definition 4.8 above. In particular, $\mathcal{F}$ is closed under (composition and) retracts (and in fact we will see that every map in $\mathcal{F}$ is a retract of a dependent projection), whereas the fibrations as defined in Theorem 6.5 are only closed under (composition and) isomorphisms.

5. A PRE-MODEL STRUCTURE ON $\mathcal{C}\ell(\mathbb{T})$

In this section we get to the core of our presentation: in presence of mapping cylinders for contexts in a dependent type theory $\mathbb{T}$, the classifying category $\mathcal{C}\ell(\mathbb{T})$ admits a pre-model structure (see Appendix A) and so it presents a homotopy theory, namely the homotopy category associated to it. This is probably the weightiest way in which we can meaningfully assert that a dependent type theory has an inherent homotopical content.

Let us now go straightly to the point of our main result.

[7] With respect to [Lum11], our definition of $\mathcal{F}_0$ (and of $\mathcal{TF}_0$ accordingly) is slightly different, as we take $\mathcal{F}_0$ to be the class of dependent projections rather than the class of display maps. Such a change is needed here as we can not replace contexts with types in our setting, due to the lack of $\Sigma-$types in our dependent type theory $\mathbb{T}$.

Theorem 5.1. [Lum11, Theorem 10] *Let $\mathbb{T}$ be dependent type theory with identity types and non-dependent mapping cylinders for contexts (cf. Definition 1.12). Then, with respect to the notations of Definition 4.8, $(\mathcal{W}, \mathcal{F}, \mathcal{C})$ is a pre-model category structure on $\mathcal{C}\ell(\mathbb{T})$. Furthermore, for such a pre-model category, $\mathcal{TF}$ and $\mathcal{TC}$ are the classes of trivial fibrations and trivial cofibrations respectively.*

We are going to break down the proof of the above Theorem in a sequence of intermediate results, which we will record as separate Propositions/Lemmas. Not all of these results, however, require $\mathbb{T}$ to have non-dependent mapping cylinders for contexts: actually, most of them are true just in presence of identity contexts. Thus, for the rest of the section, we will keep the background hypothesis that our type theory $\mathbb{T}$ is as in Blanket Assumption 1.3 and we will mention explicitly when the presence of mapping cylinders is needed. We will also use the names:

- *generating fibrations* for the maps in $\mathcal{F}_0$;
- *trivial cofibrations* for the morphisms in $\mathcal{TC}$;
- *fibrations* for the elements of $\mathcal{F}$;
- *generating trivial fibrations* for the maps in $\mathcal{TF}_0$;
- *cofibrations* for the morphisms in $\mathcal{C}$;
- *trivial fibrations* for the morphisms in $\mathcal{TF}$;
- *weak equivalences* for the maps in $\mathcal{W}$.

We start by establishing one of the required weak factorization systems.

Proposition 5.2. [GG08, Lemma 11] *Every map $f\colon \Phi \to \Psi$ in $\mathcal{C}\ell(\mathbb{T})$ has a factorization $f = p \circ i$, where $i \in \mathcal{TC} \cap \mathcal{W}$ and $p \in \mathcal{F}_0$, i.e. p is a dependent projection.*

Proof. Given an f as in the Proposition, let $\mathrm{Id}(f)$ be the context $[x:\Phi,\ y:\Psi,\ u:\mathrm{Id}_\Psi(fx,y)]$. We factor f as

$$\Phi \xrightarrow{i_f} \mathrm{Id}(f) \xrightarrow{p_f} \Psi\,,$$

where $i_f = (x, fx, 1_{fx})$ and $p_f = (y)$. Note that clearly $p_f \circ i_f = f$ and p_f is a dependent projection (to be precise, p_f would be a dependent projection if we had put $\mathrm{Id}(f) = [y:\Psi,\ x:\Phi,\ u:\mathrm{Id}_\Psi(fx,y)]$, but the order in which we declare the variables $x:\Phi,\ y:\Psi$ in $\mathrm{Id}(f)$ does not matter because none of Φ and Ψ is a dependent context relative to the other). We thus only need to show that $i_f \in \mathcal{TC} \cap \mathcal{W}$.

In order to prove that $i_f \in \mathcal{TC}$, it is enough, thanks to Remark 4.9, to show that i_f has the left lifting property with respect to all display maps. Suppose then given a commutative diagram

(19)
$$\begin{array}{ccc} \Phi & \xrightarrow{\ \tau\ } & [v:\Lambda,\ z:D(v)] \\ {\scriptstyle i_f}\downarrow & & \downarrow{\scriptstyle \pi_D} \\ \mathrm{Id}(f) & \xrightarrow[\ \gamma\ ]{} & \Lambda \end{array}$$

Now, since $\mathcal{C}\ell(\mathbb{T})$ has pullbacks along display maps (see Lemma 2.8), we have a pullback square

$$\begin{array}{ccc} [\mathrm{Id}(f),\, z{:}D[\gamma/v]] & \xrightarrow{(\gamma,\, z)} & [v{:}\Lambda,\, z{:}D(v)] \\ {\scriptstyle \pi_{D[\gamma/v]}}\downarrow & & \downarrow{\scriptstyle \pi_D} \\ \mathrm{Id}(f) & \xrightarrow[\gamma]{} & \Lambda \end{array}$$

so that there is a unique map $t\colon \Phi \to [\mathrm{Id}(f),\, z{:}D[\gamma/v]]$ such that $\pi_{D[\gamma/v]} \circ t = i_f$ and $(\gamma,\, z) \circ t = \tau$. Thus, if we can find a section s of $\pi_{D[\gamma/v]}$ such that $s \circ i_f = t$, then the composite $(\gamma,\, z) \circ s$ would be a diagonal filler for the square (19). Therefore, it is enough to show that there is a diagonal filler for every square of the form

$$\begin{array}{ccc} \Phi & \xrightarrow{\sigma} & [\mathrm{Id}(f),\, z{:}C(x,y,u)] \\ {\scriptstyle i_f}\downarrow & & \downarrow{\scriptstyle \pi_C} \\ \mathrm{Id}(f) & \xrightarrow[1_{\mathrm{Id}(f)}]{} & \mathrm{Id}(f) \end{array} \tag{20}$$

so that $C(x,y,u)$ is a dependent type relative to the context $\mathrm{Id}(f)$. The commutativity of (20) forces σ to be of the form

$$(x{:}\Phi,\ fx{:}\Psi,\ 1_{fx}{:}\mathrm{Id}_\Psi(1_{fx},1_{fx}),\ d(x){:}C(x,fx,1_{fx})) = (i_f,\ d(x){:}C(x,fx,1_{fx})),$$

so that $x{:}\Phi \vdash d(x){:}C(x,fx,1_{fx})$. Consider now the dependent context

$$\Delta(y_0,y_1,v) = [u{:}\mathrm{Id}(fx,y_0),\ z{:}C(x,y_0,u)]$$

relative to $[x{:}\Phi,\ y_0,y_1{:}\Psi,\ v{:}\mathrm{Id}(y_0,y_1)]$. We can then use Id Cxt$-$ELIM to derive the judgment

$$\frac{\begin{array}{c} x{:}\Phi,\ y_0,y_1{:}\Psi,\ v{:}\mathrm{Id}(y_0,y_1),\ \Delta(y_0,y_1,v) \vdash C(x,y_1,v\circ u){:}\mathsf{Type} \\ x{:}\Phi,\ y{:}\Psi,\ \Delta[y/y_0,y/y_1,1_y/v] \vdash z{:}C(x,y,1_y\circ u) = C(x,y,u) \end{array}}{x{:}\Phi,\ y_0,y_1{:}\Psi,\ v{:}\mathrm{Id}(y_0,y_1),\ \Delta(y_0,y_1,v) \vdash J_{y.z}(x,y_0,y_1,v){:}C(x,y_1,v\circ u)}$$

By setting $n(x,y,u,z) := J_{y.z}(x,fx/y_0,y/y_1,u/v)$ we can thus derive

$$x{:}\Phi,\ y{:}\Psi,\ u{:}\mathrm{Id}(fx,y),\ z{:}C(x,fx,1_{fx}) \vdash n(x,y,u,z){:}C(x,y,u\circ 1_{fx}).$$

Substituting $d(x)$ for z we then get

$$x{:}\Phi,\ y{:}\Psi,\ u{:}\mathrm{Id}(fx,y) \vdash n(x,y,u,d(x)){:}C(x,y,u\circ 1_{fx}).$$

Since there is a homotopy $\psi_u{:}\mathrm{Id}(u\circ 1_{fx},u)$ by Lemma 3.8, we can use Lemma 1.10 to get

$$x{:}\Phi,\ y{:}\Psi,\ u{:}\mathrm{Id}(fx,y) \vdash (\psi_u)_!(n(x,y,u,d(x))){:}C(x,y,u)$$

Setting $j := (x,y,u,(\psi_u)_!(n(x,y,u,d(x))))$, we get the required diagonal filler in (20) because clearly $\pi_C \circ j = 1_{\mathrm{Id}(f)}$ and $j \circ 1_f = (\psi_{1_{fx}/u})_!(n(x,fx/y,1_{fx}/u,d(x))) = d(x)$.

Finally, we show that $i_f \in \mathcal{W}$. We claim that $\pi_\Phi = (x)\colon\ \mathrm{Id}(f) \to \Phi$ is a two-sided quasi-inverse for i_f. Indeed, it is clear that $\pi_\Phi \circ i_f = \mathrm{id}_\Phi$, so that it is enough to find, for $x : \Phi$, $y : \Psi$ and $u : \mathrm{Id}_\Psi(fx,y)$, a path between the terms $(x,fx,1_{fx})$ and (x,y,u) of the context $\mathrm{Id}(f)$. Now, by Remark 1.11, the identity context between these two terms can be taken as

$$[v{:}\mathrm{Id}_{[\Phi,\Psi]}((x,fx),\ (x,y)),\ w{:}\mathrm{Id}(v_!(1_{fx}),u)],$$

where the second identity context is over $\mathrm{Id}_\Psi(fx,y)$. Now, a path $v : \mathrm{Id}_{[\Phi,\Psi]}((x,fx),\ (x,y))$ is of the form

$$[v_1{:}\mathrm{Id}_\Phi(x,x),\ v_2{:}\mathrm{Id}_\Psi((v_1)_!(fx),y)].$$

We can then take $v_1 := r_\Phi(x)$ and consequently we can set $v_2 := u : \mathrm{Id}_\Psi(fx, y)$ (because we have that $(r_\Phi(x))_!(fx) = fx$). So we must find a path $w : \mathrm{Id}((r_\Phi(x),\ u)_!(1_{fx}), u)$. This is a specific instance of a more general judgment we can infer.[8] Indeed, given a context Φ and $p : \mathrm{Id}_\Phi(x, x')$ (for $x, x' : \Phi$), set $\overline{p} := (r_\phi(x),\ p)$. Then we get a term in $\mathrm{Id}(\overline{p}_!(r_\phi(x)),\ p)$ by using Id Cxt−ELIM with premises

$$x, x' : \Phi,\ u : \mathrm{Id}_\Phi(x, x') \vdash \Theta(x, x', u) := \mathrm{Id}_{\mathrm{Id}_\Phi(x,x')}(\overline{p}_!(r_\phi(x)),\ p) : \mathsf{Cxt}$$

and

$$x : \Phi \vdash r_{\mathrm{Id}_\Phi(x,x)}(r_\Phi(x)) : \mathrm{Id}_{\mathrm{Id}_\Phi(x,x')}(r_\Phi(x),\ r_\Phi(x)),$$

where $\mathrm{Id}_{\mathrm{Id}_\Phi(x,x')}(r_\Phi(x),\ r_\Phi(x)) = \Theta(x/x,\ x/x',\ r_\Phi(x)/u)$ because $\overline{r_\phi(x)} = r_{[\Phi,\ \Phi]}(x, x)$ (thanks again to Remark 1.11) and we can then use the computation rule for !. (Note, en passant, that the exact same kind of strategy applies to get our paths v and w in the case we wanted to use the description of identity contexts given in the proof of Theorem 1.8). □

Remark 5.3. Note that in the proof of Proposition 5.2 above, π_Φ is a dependent projection and an equivalence, so it is in $\mathcal{F}_0 \cap \mathcal{W}$. Thus, the above Lemma, together with Lemma 5.9 below, says that we can factor each map f in $\mathcal{C}\ell(\mathbb{T})$ as a map (i_f) which is a right inverse to a trivial (generating) fibration, followed by a (generating) fibration (p_f).

Remark 5.4. Our intuitive interpretation of contexts as spaces matches very well with the content of Proposition 5.2. Indeed, that Proposition exhibits a purely syntactic counterpart of the following well-known factorization result in homotopy theory. Given a map $f : Y \to X$ of spaces, we can find a factorization of f as

$$Y \xrightarrow{i_f} P(f) \xrightarrow{p_f} X\ , \qquad f = p_f \circ i_f$$

where:

- $P(f)$ is the pullback object of $X^I \xrightarrow{\mathrm{ev}_0} X \xleftarrow{f} Y$ (where I is the closed interval $[0, 1]$ and ev_0 is the map evaluating a path in X at 0), i.e. $P(f) = \{(u, y) \in X^I \times Y : u(0) = f(y)\}$. Such a $P(f)$ is precisely our context $\mathrm{Id}(f)$ above;
- i_f is the map sending $y \in Y$ into $(c_{f(y)},\ y)$, where $c_{f(y)}$ denotes the constant path in X based at $f(y)$, and such an i_f is a homotopy equivalence. Again, this i_f is exactly the one we constructed in the proof of Proposition 5.2;
- p_f sends $(u, y) \in P(f)$ to $u(1) \in X$ and is a (Hurewicz) fibration.

With Proposition 5.2 in hand, it is easy to prove the following

Theorem 5.5. [GG08, Theorem 10] *Given a dependent type theory* $\mathbb{T}$ *satisfying the axioms* (4) *for identity types, the pair*

$$(\mathcal{TC},\ \mathcal{F})$$

is a weak factorization system on $\mathcal{C}\ell(\mathbb{T})$.

Proof. Recall that $\mathcal{TC} = {}^{\pitchfork}\mathcal{F}_0$ and $\mathcal{F} = \mathcal{TC}^{\pitchfork}$. The factorization axiom is given by Proposition 5.2 since $\mathcal{F}_0 \subseteq \mathcal{F}$. To show the weak orthogonality axiom holds for $(\mathcal{TC}, \mathcal{F})$ we only need to prove that $\mathcal{TC} = {}^{\pitchfork}\mathcal{F}$, since $\mathcal{F} = \mathcal{TC}^{\pitchfork}$ by definition. Certainly ${}^{\pitchfork}\mathcal{F} \subseteq \mathcal{TC}$, again because $\mathcal{F}_0 \subseteq \mathcal{F}$ and ${}^{\pitchfork}(-)$ (as well as $(-)^{\pitchfork}$) reverses inclusions of classes of maps in $\mathcal{C}\ell(\mathbb{T})$. To show that also the other inclusion holds, we observe that every map in $\mathcal{TC}$ has, by definition, the LLP with respect to every dependent projections. But now every map in $\mathcal{F}$ is a retract of a dependent projection, thanks to Proposition 5.2 and to the Retract Argument (see Appendix A). This implies that $\mathcal{TC} \subseteq {}^{\pitchfork}\mathcal{F}$ and we are done. □

[8] Credits go to Chris Kapulkin in this part, as he was the *deus ex machina* here.

Next, we take care of the other factorization system: here is where mapping cylinders come into play.

Lemma 5.6. [Lum11, Lemma 15] *Let $\mathbb{T}$ be a dependent type theory with identity types and non-dependent mapping cylinders for contexts. Then every map in $\mathcal{C}\ell(\mathbb{T})$ factors as a map in $\mathcal{C}$ followed by a map in $\mathcal{TF}_0$.*

Proof. Given a context morphism $f\colon \Theta \to \Delta$, we factor it through its mapping cylinder as

$$\Theta \xrightarrow{(f,\mathsf{in\text{-}top})} [\Delta, \mathsf{Cyl}_f] \xrightarrow{\pi_{\mathsf{Cyl}_f}} \Delta$$

Let us first show that $(f, \mathsf{in\text{-}top})$ is a cofibration. We need to find a dotted filler in every solid diagram of the form

$$\begin{array}{ccc} \Theta & \xrightarrow{h} & [\Theta, A] \\ {\scriptstyle (f,\mathsf{in\text{-}top})}\downarrow & \overset{t}{\nearrow} & \downarrow{\scriptstyle \pi_A} \\ [\Delta, \mathsf{Cyl}_f] & \xrightarrow[k]{} & \Theta \end{array}$$

Since the first component of every such a dotted filler is forced to be k, we only need to derive a dependent term

$$y\!:\!\Delta,\ z\!:\!\mathsf{Cyl}_f(y) \vdash t(y,z)\!:\!A(k(y,z)),$$

such that $t(f(x), \mathsf{in\text{-}top}(x)) = h(x)$. Let (a, α) be such that $\Gamma \vdash (a, \alpha)$ contract A. Then we can get our desired term via a single application of Cyl−ELIM with premises:

- $y\!:\!\Delta,\ z\!:\!\mathsf{Cyl}_f(y) \vdash A(k(y,z))\!:\!\mathsf{Type}$;
- for $y\!:\!\Delta$, $d_{\mathsf{base}}(y) := a(k(y, \mathsf{in\text{-}base}(y)))\!:\!A(k(y, \mathsf{in\text{-}base}(y)))$;
- for $x\!:\!\Theta$, $d_{\mathsf{top}}(x) := h(x)\!:\!A(k(fx, \mathsf{in\text{-}top}(x)))$;
- for $x\!:\!\Theta$,

  $$d_{\mathsf{cyl}}(x) := \alpha(x,\ \mathsf{in\text{-}cyl}(x)_!(d_{\mathsf{top}}(x)),\ a(k(fx, \mathsf{in\text{-}base}(fx)))$$

  which is a term of type

  $$\mathrm{Id}_{A(k(fx,\mathsf{in\text{-}base}(fx)))}(\mathsf{in\text{-}cyl}(x)_!(d_{\mathsf{top}}(x)),\ a(k(fx, \mathsf{in\text{-}base}(fx)))).$$

We now need to show that $y : \Delta \vdash \mathsf{isContr}(\mathsf{Cyl}_f(y))$ in order to get that $\pi_{\mathsf{Cyl}_f} \in \mathcal{TF}_0$. We take $y\!:\!\Delta \vdash \mathsf{in\text{-}base}(y)\!:\!\mathsf{Cyl}_f(y)$ as a canonical inhabitant (see Definition 4.4). Since we can take inverses and composite of paths, to get a contraction $y\!:\!\Delta,\ z, z'\!:\!\mathsf{Cyl}_f(y) \vdash \alpha(y, z, z')\!:\!\mathrm{Id}(z, z')$ it is enough to derive a judgment

$$y\!:\!\Delta,\ z\!:\!\mathsf{Cyl}_f(y) \vdash \beta(y,z)\!:\!\mathrm{Id}(z, \mathsf{in\text{-}base}(y)).$$

Again, we can get such a dependent term via Cyl−ELIM with premises:

- $y\!:\!\Delta,\ z\!:\!\mathsf{Cyl}_f(y) \vdash \mathrm{Id}_{\mathsf{Cyl}_f(y)}(z, \mathsf{in\text{-}base}(y))$;
- $y\!:\!\Delta \vdash d_{\mathsf{base}}(y) := r(\mathsf{in\text{-}base}(y))\!:\!\mathrm{Id}(\mathsf{in\text{-}base}(y),\ \mathsf{in\text{-}base}(y))$;
- $x\!:\!\Theta \vdash d_{\mathsf{top}}(x) := \mathsf{in\text{-}cyl}(x)\!:\!\mathrm{Id}(\mathsf{in\text{-}top}(x),\ \mathsf{in\text{-}base}(fx))$;
- for $x\!:\!\Theta$, the needed term $d_{\mathsf{cyl}}(x)$ of context

  $$\mathrm{Id}_{\mathrm{Id}(\mathsf{in\text{-}base}(fx),\ \mathsf{in\text{-}base}(fx))}(\mathsf{in\text{-}cyl}(x)_!(\mathsf{in\text{-}cyl}(x)),\ r(\mathsf{in\text{-}base}(fx)))$$

  can be obtained via a suitable substitution in a more general judgment

  $$x, x'\!:\!\Gamma,\ u\!:\!\mathrm{Id}(x, x') \vdash \mathsf{transport\text{-}lemma}(u)\!:\!\mathrm{Id}(u_!(u),\ r(x_1))$$

  which follows easily from Id Cxt-ELIM as, when $u = r_\Gamma(x)$, we can simply take $r(r_\Gamma(x))$: $\mathrm{Id}(r_\Gamma(x), r_\Gamma(x))$.

$\square$

Remark 5.7. Type-theoretic mapping cylinders and the content of Lemma 5.6 can be seen, respectively, as the syntactic counterpart of mapping cylinders for maps of spaces and the factorization they induce. Indeed, consider a map $f\colon X \to Y$ between topological spaces. In classical homotopy theory, one defines the mapping cylinder of such a map as the pushout (object) of

$$X \times I \xleftarrow{j_0} X \xrightarrow{f} Y,$$

where j_0 is the inclusion of X at the bottom of the cylinder $X \times I$ (where I is the closed interval $[0,1]$). Call such a pushout M_f. By the universal property of the pushout, the maps

$$X \times I \xrightarrow{p_I} X, \qquad Y \xrightarrow{\mathrm{id}_Y} Y$$

(where p_I is the projection map) induce a unique map $r_f\colon M_f \to Y$ compatible with the canonical coprojections $X \times I \to M_f$ and $Y \to M_f$ into the pushout. Now, for a point $y \in Y$, the (homotopy) fiber $r_f^{-1}(\{y\})$ can be thought of as representing our $\mathsf{Cyl}_f(y)$. The canonical inclusion $k_Y\colon Y \to M_f$ represents the $\mathsf{in\text{-}base}$ constructor (for $y \in Y$, $k_Y(y) \in r_f^{-1}(\{y\})$), whereas the composite map $X \to X \times I \to M_f$, where $X \to X \times I$ is the inclusion of X at the top of the cylinder over X and $X \times I \to M_f$ is the canonical map into the pushout, can be considered as the $\mathsf{in\text{-}top}$ constructor (for $x \in X$, $k_f \in r_f^{-1}(\{fx\})$). Given $x \in X$, there is a canonical path between $k_f(x)$ and $k_Y(fx)$ obtained by composing the path $t \mapsto (x,t)$ in $X \times I$ with the quotient map $X \times I \to M_f$ (this is because fx and $(x,0)$ get identified in M_f), so that such a path can be understood as the constructor $\mathsf{in\text{-}cyl}(x)$. The analogue of Lemma 5.6 for spaces is the well-known factorization of each map into a (closed Hurewicz) cofibration followed by a homotopy equivalence (a strong deformation retract). More precisely, given a map $f\colon X \to Y$, f admits the factorization

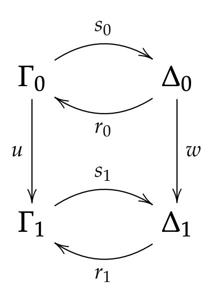
$$X \xrightarrow{k_f} M_f \xrightarrow{r_f} Y, \qquad r_f \circ k_f = f,$$

where k_f and r_f are as above.

Next, we must show that $\mathcal{W}$ is indeed a good candidate to be the class of weak equivalences for a pre-model category structure.

Lemma 5.8. [Lum11, Lemma 16] *The class $\mathcal{W}$ has the 2-out-of-3 property and is closed under retracts.*

Proof. Since we can compose and take inverses of paths and homotopies, we can whisker a homotopy with a path (see the proof of Lemma 3.8) and we can apply context morphisms to paths and homotopies (see Lemma 3.11), this follows mostly formally. For example suppose we have a retraction of $w \in \mathcal{W}$ as in



Now, if $i\colon \Delta_1 \to \Delta_0$ is a left quasi-inverse for w, then $r_0 \circ i \circ s_1$ is a left quasi-inverse for u because

$$r_0 \circ i \circ s_1 \circ u = r_0 \circ i \circ w \circ s_0 \simeq r_0 \circ s_0 = \mathrm{id},$$

where we used the symbol " $\simeq$ " to mean that there is a path from the left hand side to the right hand side. Similarly, given a right quasi-inverse $j\colon \Delta_1 \to \Delta_0$ for w, then $r_0 \circ j \circ s_1$ is a right quasi-inverse for u because

$$u \circ r_0 \circ j \circ s_1 = r_1 \circ w \circ j \circ s_1 \simeq r_1 \circ s_1 = \mathrm{id}\,.$$

As for the 2-out-of-3 property, suppose we have maps $w\colon \Delta_0 \to \Delta_1$ and $w'\colon \Delta_1 \to \Delta_2$ with w and $w'w$ being equivalences. If l is a left quasi inverses for $w'w$ and r is a right quasi-inverse for w, then wl is a left quasi-inverse for w', as we have

$$wlw' \simeq wlw'ws \simeq ws = \mathrm{id},$$

where the various paths involved are given by applying wl and w to suitable instances of the paths witnessing that l is a left quasi-inverse for $w'w$ and that s is a right quasi-inverse for w. Clearly, if r is a right quasi-inverse of $w'w$, then wr is a right quasi-inverse for w'. The other cases (where w, w' are equivalences or where $w'w$ and w' are equivalences) follow similarly. □

Lemma 5.9. [Lum11, Lemma 17] $\mathcal{TF}_0 = \mathcal{F}_0 \cap \mathcal{W}$.

Proof. We have to show that, for any dependent context $\Gamma \vdash \Delta : \mathsf{Cxt}$, the dependent projection $\pi_{\Gamma,\Delta}$ is an equivalence if and only if $\Gamma \vdash \mathsf{isContr}(\Delta)$.

Suppose first that $\Gamma \vdash (a, \alpha)\ \mathsf{contract}\ \Delta$. Then $(x, a(x))$ is a morphism $\Gamma \to [\Gamma, \Delta]$ which gives a right inverse for π_Δ. A path between $(x, a(x))$ and (x, y) for $x : \Gamma$ and $y : \Delta$ is given by $(r_\Gamma(x),\ \alpha(x, a(x), y))$ which is a term of

$$\mathrm{Id}_{[\Gamma,\Delta]}((x, a(x)),\ (x, y)) = [u : \mathrm{Id}_\Gamma(x, x),\ \mathrm{Id}_\Delta(u_!(a(x)),\ y)]$$

(see Remark 1.11).

Conversely, suppose π_Δ is an equivalence with right quasi-inverse given by $g\colon \Gamma \to [\Gamma, \Delta]$, so that $g = (g_\Gamma,\ g_\Delta)$ with

$$x : \Gamma \vdash \epsilon_x : \mathrm{Id}_\Gamma(g_\Gamma(x), x), \quad x : \Gamma \vdash g_\Delta(x) : \Delta(g_\Gamma(x)),$$

for some suitable dependent path ϵ witnessing that g is a right quasi-inverse for f. Then $a(x) := \epsilon(x)_!(g_\Delta(x))$, gives us a canonical inhabitant for $\Delta(x)$. We thus just need to find a contraction

$$x : \Gamma,\ y, y' : \Delta \vdash \alpha(x, y, y') : \mathrm{Id}(y, y'),$$

which amounts to derive a judgment

$$x : \Gamma,\ y : \Delta \vdash \beta(x, y) : \mathrm{Id}(a(x), y).$$

By Remark 4.3 and the hypothesis that π_Δ is an equivalence, we have that any right quasi-inverse for π_Δ is also a left-quasi inverse. Thus, we have a judgment

$$x : \Gamma,\ y : \Delta \vdash \eta_{(x,y)} : \mathrm{Id}_{[\Gamma,\Delta]}((x, a(x)),\ (x, y))$$

Using Remark 1.11, we see that $\eta_{(x,y)} = (\eta^\Gamma_{(x,y)},\ \eta^\Delta_{(x,y)})$, where

$$\eta^\Gamma_{(x,y)} : \mathrm{Id}_\Gamma(x, x), \quad \eta^\Delta_{(x,y)} : \mathrm{Id}_\Delta((\eta^\Gamma_{(x,y)})_!(a(x)),\ y).$$

Applying $\mathsf{dep\text{-}cong}$ we can now derive

$$x : \Gamma,\ y : \Delta \vdash \mathsf{dep\text{-}cong}(x.a(x);\ \eta^\Gamma_{(x,y)}) : \mathrm{Id}_\Delta((\eta^\Gamma_{(x,y)})_!(a(x)),\ a(x))$$

and so, the composite of such a path with $(\eta^\Delta_{(x,y)}))^{-1}$ gives us the desired path $\beta(x, y)$. □

With what we proved up to now, we can therefore conclude that:

- $(\mathcal{TC},\ \mathcal{F}))$ and $(\mathcal{C},\ \mathcal{TF})$ are weak factorization systems (the first pair forms a weak factorization system by Theorem 5.5, whereas the second one is a weak factorization system thanks to Lemma 5.6 and the fact that $\mathcal{TF} = \mathcal{C}^\pitchfork$ and $\mathcal{C} = {}^\pitchfork\mathcal{TF}$, which can be proven by an argument similar to the one used in the proof of Theorem 5.5);

- $\mathcal{W}$ is closed under retracts and has the 2-out-of-3 property (Lemma 5.8), i.e. $\mathcal{W}$ satisfies necessary properties to be the class of weak-equivalences for a pre-model category structure on $\mathcal{C}\ell(\mathbb{T})$;
- $\mathcal{TF}_0 = \mathcal{F}_0 \cap \mathcal{W}$ (Lemma 5.9).

Thus, Theorem 5.1 is proved as soon as we can show that $\mathcal{TF} = \mathcal{F} \cap \mathcal{W}$ and $\mathcal{TC} = \mathcal{C} \cap \mathcal{W}$. Consequently, the reader will probably not be so surprised by the content of the following

Lemma 5.10. [Lum11, Lemma 18] *$\mathcal{TF} = \mathcal{F} \cap \mathcal{W}$ and $\mathcal{TC} = \mathcal{C} \cap \mathcal{W}$.*

Proof. The proof is essentially formal. First of all, by definition we have $\mathcal{TF}_0 \subseteq \mathcal{F}_0$, so, since both ${}^{\pitchfork}(-)$ and $(-)^{\pitchfork}$ reverse inclusions between classes of maps in $\mathcal{C}\ell(\mathbb{T})$, $\mathcal{TC} \subseteq \mathcal{C}$ and then $\mathcal{TF} \subseteq \mathcal{F}$. We are now going to prove some other inclusions which will give us the thesis.

(a) $\mathcal{TC} \subseteq \mathcal{W}$: if $\mathcal{TC} = {}^{\pitchfork}(\mathcal{F}_0)$, we can factor it as $f = pi$, where $i \in \mathcal{C} \cap \mathcal{W}$ and $p \in \mathcal{F} = \mathcal{TC}^{\pitchfork}$. Thus, f has the left lifting property with respect to p, so it is a retract of i by the Retract Argument (see Appendix A). In particular, $i \in \mathcal{W}$.

(b) $\mathcal{W} \cap \mathcal{C} \subseteq \mathcal{TC}$: again, given $f \in \mathcal{C} \cap \mathcal{W}$, write $f = pi$ where $i \in \mathcal{TC}$ and $p \in \mathcal{W}$. By part (a) above and by the 2-out-of-3 property for $\mathcal{W}$, $p \in \mathcal{W}$ and then $p \in \mathcal{F}_0 \cap \mathcal{W} = \mathcal{TF}_0$. Therefore f has the left lifting property with respect to p and is then a retract of i, which implies that $f \in \mathcal{TC}$, because $\mathcal{TC}$ is closed under retracts.

(c) $\mathcal{TF} \subseteq \mathcal{W}$: this is dual to (a).

(d) $\mathcal{W} \cap \mathcal{F} \subseteq \mathcal{TF}$: as usual, consider a factorization $f = pi$ where $p \in \mathcal{F}_0$ and $i \in \mathcal{C} \cap \mathcal{W}$. Then f has the right lifting property with respect to i, so it is a retract of p which must then be a weak equivalence. Therefore, $p \in \mathcal{W} \cap \mathcal{F}_0 = \mathcal{TF}_0$ and hence $f \in \mathcal{TF}$, by definition of $\mathcal{TF}$.

□

The proof of Theorem 5.1 is now complete.

Remark 5.11. For each context Γ, the unique map $\Gamma \to [\,]$ is a dependent projection, so that each object in $\mathcal{C}\ell(\mathbb{T})$ is fibrant with respect to the pre-model category structure given by Theorem 5.1.

We end this Section by providing a characterization of the maps in $\mathcal{TC}$.

Definition 5.12. [GG08, Section 5] A context morphism $f\colon \Phi \to \Psi$ in $\mathcal{C}\ell(\mathbb{T})$ is a *type-theoretic injective equivalence* if there is a context morphism $s\colon \Psi \to \Phi$ such that the following judgments are inferable in $\mathbb{T}$:

$$x : \Phi \vdash s(f(x)) = x : \Phi, \tag{21}$$

$$y : \Psi \vdash \epsilon_y : \mathrm{Id}_\Psi(f(s(y)), y), \tag{22}$$

$$x : \Phi \vdash \epsilon_{fx} = 1_{fx} : \mathrm{Id}_\Psi(fx, fx). \tag{23}$$

We can read (21) and (22) as saying that s is a (strict) left inverse for f and that s is a right quasi-inverse of f respectively (see Definition 4.1 above). Moreover, (23) implies in particular that $x : \Phi \vdash f(s(fx)) = fx : \Psi$.

Proposition 5.13. [GG08, Lemma 13] *A context morphism $f\colon \Phi \to \Psi$ is in $\mathcal{L}$ if and only if f is a type-theoretic injective equivalence.*

Proof. We define $(\mathcal{TC})'$ to be the class of all context morphisms $f\colon \Phi \to \Psi$ such that there is a dotted filler t in every solid diagram of the form

$$
\begin{array}{ccc}
\Phi & \xrightarrow{i_f} & \mathrm{Id}(f) \\
{\scriptstyle f}\big\downarrow & \overset{t}{\nearrow} & \big\downarrow{\scriptstyle p_f} \\
\Psi & \xrightarrow[1_\Psi]{} & \Psi
\end{array}
\tag{24}
$$

where $f = p_f i_f$ is the factorization of f given by Proposition 5.2. Note that every such map $t = (t_1(y)\colon\Phi, t_2(y) = y\colon\Psi, t_3(y) : \mathrm{Id}(f(t_1(y)), t_2(y)))$ (where $y\colon\Psi$) gives an $s\colon \Psi \to \Phi$ satisfying (21)-(23) by setting $s := t_1$ and $\epsilon_y := t_3(y)$. Viceversa, every type-theoretic injective equivalence $s\colon \Psi \to \Phi$ provides a filler t for the above diagram, by reading the previous definitions from right to left. Thus, in order to get our result, we need to show precisely that $(\mathcal{TC})' = \mathcal{TC}$. The inclusion $\mathcal{TC} \subseteq (\mathcal{TC})'$ is immediate from the fact that $p_f \in \mathcal{R}$ for each f. As for the other inclusion, suppose we have a filler t in (24) for $f\colon \Phi \to \Psi$. Then f is a retract of i_f and $i_f \in \mathcal{TC}'$, so also f is. ☐

6. The fibration-category structure of $\mathcal{C}\ell(\mathbb{T})$

The outstanding achievement of turning $\mathcal{C}\ell(\mathbb{T})$ into a pre-model category came in the last section at the price of requiring our dependent type theory to admit constructors for mapping cylinders other than for identity types. However, as we have already remarked, not the whole pre-model category structure relied on the presence of mapping cylinders. Namely, we could deduce the following results for $\mathcal{C}\ell(\mathbb{T})$ just by exploiting the syntactic power of identity contexts:

- every map in $\mathcal{C}\ell(\mathbb{T})$ factors as an equivalence (which is also a trivial cofibration) followed by a dependent projection (and this was the factorization axiom for a specific weak factorization system);
- the class $\mathcal{W}$ of equivalences is closed under retracts and has the 2-out-of-3 property;
- the dependent projections which are equivalences are exactly the dependent projections $\pi_\Delta\colon [\Gamma,\ \Delta] \to \Gamma$, where Δ is a contractible context.

There is a way to abstract from the setting of classifying categories and define a structure on arbitrary categories which comprises the specific case of $\mathcal{C}\ell(\mathbb{T})$ for a dependent type theory with axioms for identity type. We spoil immediately all the surprise and reveal what kind of structure we will consider.

Definition 6.1. [AKL15, Definition 3.2.1] A *fibration category* is a category $\mathcal{C}$ equipped with a pair of subclasses of morphisms in $\mathcal{C}$

$$(\mathcal{W},\ \mathcal{F}),$$

where the maps in $\mathcal{W}$ are called *weak equivalences* and the maps in $\mathcal{F}$ are called *fibrations*, satisfying the following properties.

(1) $\mathcal{W}$ satisfies the *2-out-of-6* property, i.e. given composable maps

$$W \xrightarrow{f} X \xrightarrow{g} Y \xrightarrow{h} Z,$$

if gf and hg are in $\mathcal{W}$, then so are f, g, h and hgf.

(2) $\mathcal{F}$ is closed under composition.

(3) All isomorphisms are *trivial fibrations*, that is, all isomorphisms are both in $\mathcal{W}$ and in $\mathcal{F}$.

(4) $\mathcal{C}$ has a terminal object 1.

(5) Pullbacks along fibrations exist in $\mathcal{C}$. Fibrations and trivial fibrations are stable under pullbacks.

(6) Every object in $\mathcal{C}$ is *fibrant*, that is, for all objects X in $\mathcal{C}$, the unique map $X \to 1$ is a fibration.
(7) For any object X of $\mathcal{C}$, there is a factorization

$$X \to PX \to X \times X$$

of the diagonal map $\Delta\colon X \to X \times X$ into a weak equivalence followed by a fibration. (Note that, for any $X \in \mathcal{C}$, the product $X \times X$ exists thanks to axioms (5) and (6) above).

Remark 6.2. The 2-out-of-6 property implies the 2-out-of-3 property for $\mathcal{W}$. This is easily seen by considering the special cases where, given composable arrows f, g, h as in Definition 6.1, both gf and hg are identity morphisms or at least one among f, g and h is an identity morphism

Here is a way to get a lot of examples of fibration categories.

Proposition 6.3. *Let $(\mathcal{M}, \mathcal{W}, \mathcal{F}, \mathcal{C})$ be a model category. Let $\mathcal{M}_f$ be the full subcategory of $\mathcal{M}$ spanned by the fibrant objects in $\mathcal{M}$ (which are, as in Definition 6.1 above, the objects $A \in \mathcal{M}$ for which the unique map $A \to 1$ is in $\mathcal{F}$). Then*

$$(\mathcal{M}_f,\ \mathcal{W} \cap \mathcal{M}_f,\ \mathcal{F} \cap \mathcal{M}_f)$$

is a fibration category.

Proof. Let us take a look at the various properties of Definition 6.1 that one has to check.

(1) The weak equivalences in a model category satisfy the 2-out-of-3 property and are *saturated*, i.e. a map $f\colon X \to Y$ is a weak equivalence if and only if its image in $\mathrm{Ho}(\mathcal{M})$ is an isomorphism (see [Hov99, Theorem 1.2.10]). This implies that $\mathcal{W}$ (and therefore also $\mathcal{W} \cap \mathcal{M}_f$) has the 2-out-of-6 property. For, let

$$A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$$

be composable morphisms in $\mathcal{M}$ with gf and hg weak equivalences. If $\gamma\colon \mathcal{M} \longrightarrow \mathrm{Ho}(\mathcal{M})$ is the localization functor, we have that $\gamma(gf)$ and $\gamma(hg)$ are isomorphism sin $\mathrm{Ho}(\mathcal{M})$. Since $\gamma(f)(\gamma(gf))^{-1}$ and $(\gamma(hg))^{-1}\gamma(h)$ are a right and a left inverse for $\gamma(g)$ respectively, g is a weak equivalence in $\mathcal{M}$. Using the two-out-of-three property, we then get that also f, h and hgf are weak equivalences, as required.
(2) This is obvious because maps in $\mathcal{F}$ have the right lifting property with respect to maps in $\mathcal{C} \cap \mathcal{W}$.
(3) In a model category isomorphisms are weak equivalences and they certainly are fibrations.
(4) A model category has a terminal object which is in $\mathcal{M}_f$ and is the terminal object for it (because $\mathcal{M}_f$ is a full subcategory of $\mathcal{M}$).
(5) The pullback of a fibration between fibrant objects in $\mathcal{M}$ along a map between fibrant objects is still a fibration between fibrant objects so, in particular, $\mathcal{M}_f$ has all pullbacks along fibrations in $\mathcal{M}_f$ (we use again the fact that $\mathcal{M}_f$ is a full subcategory of $\mathcal{M}$ to be sure that pullbacks as computed in $\mathcal{M}$ are also pullbacks in $\mathcal{M}_f$) and fibrations in $\mathcal{M}_f$ are stable under pullbacks. Acyclic fibrations in $\mathcal{M}$ are stable under pullbacks because they are defined via a right lifting property and therefore they are stable under pullbacks also in $\mathcal{M}_f$.
(6) This is clear from the definition of $\mathcal{M}_f$.
(7) Given an object $X \in \mathcal{M}_f$ we can factor the diagonal map $\Delta\colon X \to X \times X$ as a weak equivalence followed by a fibration in $\mathcal{M}$. This is also a factorization as a weak equivalence followed by a fibration in $\mathcal{M}_f$, by definition of weak equivalences and fibrations in $\mathcal{M}_f$.

□

Remark 6.4. From the above Proposition we get in particular that the model structures on Top and on sSet given in Example 7.8 and in Example 7.9 (see Appendix A) induce a fibration category structure on Top (because every object is fibrant in Top) and on Kan (the full subcategory of sSet spanned by Kan complexes) respectively.

We are now ready to explain the fibration category structure on $\mathcal{C}\ell(\mathbb{T})$.

Theorem 6.5. [AKL15, Definition 3.2.3 and Theorem 3.2.5] *Let $\mathcal{C}\ell(\mathbb{T})$ be the classifying category of a dependent type theory with identity types. Then, with respect to the notations of Definition 4.8,*

$$(\mathcal{C}\ell(\mathbb{T}),\ \mathcal{W},\ \mathcal{F}')$$

is a fibration category.

As for Theorem 5.1, we are going to divide the proof of Theorem 6.5 in several steps. Actually, we have already shown that most of the axioms of a fibration category hold for $(\mathcal{C}\ell(\mathbb{T}),\ \mathcal{W},\ \mathcal{F}')$. Recall first of all that $\mathcal{F}'$ is the closure of the class $\mathcal{F}_0$ of dependent projections under isomorphisms (in the arrow category of $\mathcal{C}\ell(\mathbb{T})$), so $\mathcal{F}' \subseteq \mathcal{F}$ (again with respect to Definition 4.8). Now, we already know that $\mathcal{C}\ell(\mathbb{T})$ has a terminal object (the empty context) and certainly $\Gamma \to [\ ]$ is a perfectly legitimate dependent projection for each context Γ. Also, each isomorphism in $\mathcal{C}\ell(\mathbb{T})$ is certainly an equivalence and it is isomorphic to a dependent projections, namely the identity context morphism on its domain. Since $\mathcal{F}'$ is (by definition) closed under composition, these observations take care of properties (2), (3), (4) and (6) for $(\mathcal{C}\ell(\mathbb{T}),\ \mathcal{W},\ \mathcal{F}')$. Axiom (7) was also already proven in wider generality in Proposition 5.2, where we showed that *every* map $f\colon \Phi \to \Psi$ in $\mathcal{C}\ell(\mathbb{T})$ factors as a weak equivalence followed by a fibration. Thus, we only need to validate axioms (1) and (7) of Definition 7.5. The following result takes care of the first axiom

Lemma 6.6. [AKL15, Lemma 3.2.6] *$\mathcal{W}$ satisfies the 2-out-of-6 property.*

Proof. Suppose given composable maps f, g and h in $\mathcal{C}\ell(\mathbb{T})$ and suppose hg and gf are in $\mathcal{W}$. By Remark 4.3, we can assume that hg and gf comes with two-sided quasi-inverses given by $(hg)^{-1}$ and $(gf)^{-1}$ respectively. Then, reasoning in the exact same way as in the proof of Lemma 5.8, we can conclude by direct inspection that

- $(gf)^{-1}g(hg)^{-1}$ is a two-sided quasi-inverse for hgf;
- $(hg)^{-1}h$ and $f(gf)^{-1}$ are a left and a right quasi-inverse for g respectively;
- $(gf)^{-1}g$ gives a quasi-inverse for f;
- $g(hg)^{-1}$ is a quasi-inverse for h.

□

We showed in Corollary 2.9 that pullbacks along dependent projections exist in $\mathcal{C}\ell(\mathbb{T})$. In order to take care of the second half of property (5) in Definition 6.1 for $\mathcal{C}\ell(\mathbb{T})$, we first need another characterization of the class $\mathcal{W}$ of weak equivalences in terms of contractibility.

Definition 6.7. Let $f\colon \Phi \to \Psi$ be a context morphism in $\mathcal{C}\ell(\mathbb{T})$. The *homotopy fiber* of f over $y\colon\Psi$ is the context

$$\mathsf{hfib}(f, y) := [x\colon X,\ \mathrm{Id}_{\Psi}(fx, y)] \tag{25}$$

Proposition 6.8. *For a context morphism $f\colon \Phi \to \Psi$ the following are equivalent:*

(1) f is an equivalence in the sense of Definition 4.1;
(2) for each $y\colon\Psi$, the homotopy fiber of f at y is contractible (see Definition 4.4).

Proof. See [Pro13, Section 4.2-4.4]. □

The last little piece we need is the following

Lemma 6.9. [AKL15, Lemma 3.2.8] *Let $\pi_B\colon [\Gamma, B] \to \Gamma$ be a display map (see Definition 2.4). Then, for any $a:\Gamma$, the type $B(a)$ (seen as a context of length 1) and $\mathsf{hfib}(\pi_B, a)$ are equivalent.*

Proof. Fix $a:\Gamma$. We define a map

$$\phi\colon B(a) \to \mathsf{hfib}(\pi_B, a), \quad \phi(b) := ((a,b),\ r_A(a)).$$

The proposed two-sided quasi-inverse for ϕ is the map $\psi\colon \mathsf{hfib}(\mathsf{pr}_B, a) \to B(a)$ which is defined by sending $((a',b),\ p) : \mathsf{hfib}(\mathsf{pr}_B, a)$ (for $a' : \Gamma$, $b : B(a')$ and $p : \mathrm{Id}_A(a, a')$) to $p_!(b) : B(a)$ (see Eq. (14)). Now, $\psi \circ \phi = \mathrm{id}$ by the computation rule for !. Given $((a',b),\ p) : \mathsf{hfib}(\mathsf{pr}_1, a)$, a path from $\phi(\psi((a',b),\ p))) = ((a, p_!(b)),\ r_A(a))$ to $((a',b),\ p)$ can be obtained by Id Cxt$-$ELIM since, when $p = r_A(a)$, $((a, p_!(b)),\ r_A(a)) = ((a,b),\ r_A(a))$, so we can just take $r_{\mathsf{hfib}(\mathsf{pr}_B,a)}(((a,b),\ r_A(a)))$. □

Note that we can interpret the content of the above Lemma by saying that, for a display map, the fiber and the homotopy fiber over any term of the base type are equivalent. This is the syntactic counterpart of a result in the homotopy theory of spaces which says that, given a fibration $p\colon E \to B$, the fiber and the homotopy fiber over any point $b \in B$ are weakly equivalent.

Finally, we can end our proof that axiom (5) for fibration categories holds in $\mathcal{C}\ell(\mathbb{T})$ with the following

Lemma 6.10. [AKL15, Lemma 3.2.9] *The maps in $\mathcal{F}'$ and $\mathcal{F}' \cap \mathcal{W}$ are stable under pullbacks.*

Proof. By definitions of $\mathcal{F}'$, it is enough to show that display maps are stable under pullbacks and that the pullback of a display map which is a weak equivalence is still a weak equivalence. The first half is clear from 2.8. As for the second half, suppose that $\pi := \pi_B\colon [\Gamma, B] \to \Gamma$ is in $\mathcal{F}' \cap \mathcal{W}$ and $f\colon \Gamma' \to \Gamma$ is a context morphism. Denote the pulled-back display map $[\Gamma', B[f]] \to \Gamma'$ with $f^*\pi$. Then, using Lemma 6.9, we have, for $x':\Gamma'$ equivalences

$$\mathsf{hfib}(f^*\pi, x') \simeq B(fx') \simeq \mathsf{hfib}(\pi, fx').$$

By hypothesis, $\mathsf{hfib}(\pi, fx')$ is contractible. It follows that also $\mathsf{hfib}(f^*\pi, x')$ is contractible, i.e. $f^*\pi$ is a weak equivalence, as required. □

This completes the proof of Theorem 6.5.

We close our work with another interesting property of the fibration category structure on $\mathcal{C}\ell(\mathbb{T})$, which well deserves the name of *right properness*.

Lemma 6.11. [Bro73, Lemma 2][9] *The pullback of a weak equivalence in $\mathcal{C}\ell(\mathbb{T})$ along a fibration is again a weak equivalence.*

Proof. Let $p\colon E \to B$ be a fibration (a map in $\mathcal{F}'$) and $u\colon B' \to B$ a weak equivalence in $\mathcal{C}\ell(\mathbb{T})$. Now, by Remark 5.3, we can factor $u = p_u i_u$, where i_u is a weak equivalence and a right inverse to a trivial fibration, while p_u is a fibration which must be a trivial fibration by the 2-out-of-3 property for weak equivalences (see Remark 6.2). Since pullbacks of trivial fibrations are still such, we can suppose that u is itself a right inverse to a trivial fibration $v\colon B \to B'$. Consider now the pullback

[9] We are grateful to Chris Kapulkin for having pointed out this reference to us.

square

$$\begin{array}{ccc} E_1 := E \times_{B'} B & \longrightarrow & B \\ \Big\downarrow {\scriptstyle \mathsf{pr}_E} & & \Big\downarrow {\scriptstyle v} \\ E & \xrightarrow[vp]{} & B' \end{array}$$

Note that the map $\mathsf{pr}_E \colon E_1 \to E$ is a trivial fibration because it is the pullback along vp of the trivial fibration v. Now, the map $f = (p, \mathrm{id}_E) \colon E \to E_1$ is a fibration and a weak equivalence because it is a right inverse of pr_E. It follows that in the diagram

$$\begin{array}{ccc} B' \times_B E & \xrightarrow{u^* f} & B' \times_B E_1 \\ \Big\downarrow & & \Big\downarrow {\scriptstyle \mathsf{pr}_{E_1}} \\ E & \xrightarrow[f]{} & E_1' \end{array}$$

the horizontal maps are weak equivalences. By the 2-out-of-3 property, we then only need to shot that pr_{E_1} is a weak equivalence. But this follows immediately because, under the natural isomorphism

$$B' \times_B E_1 = B' \times_B (B \times_{B'} E) \cong E$$

pr_{E_1} gets identified with a map $E \to E_1$ which is a right inverse to the trivial fibration pr_E. □

7. Appendix A

Definition 7.1. Let $\mathcal{D}$ be a category.

(1) Let $i \colon A \to B$ and $p \colon X \to Y$ be maps in $\mathcal{D}$. If, for every commutative solid diagram of the form

$$\begin{array}{ccc} A & \xrightarrow{f} & X \\ {\scriptstyle i} \Big\downarrow & \nearrow^{k} & \Big\downarrow {\scriptstyle p} \\ B & \xrightarrow[g]{} & Y \end{array}$$

there is a dotted filler $k \colon B \to X$, we say that i has the *left lifting property (LLP)* with respect to p and that p has the *right lifting property (RLP)* with respect to i.

(2) Given a class $\mathcal{A}$ of morphisms in $\mathcal{D}$ we denote with $\mathcal{A}^{\pitchfork}$ the class of morphisms in $\mathcal{D}$ having the RLP with respect to every map in $\mathcal{A}$. Similarly, we denote with ${}^{\pitchfork}\mathcal{A}$ the class of morphisms in $\mathcal{D}$ having the LLP with respect to every map in $\mathcal{A}$.

Definition 7.2. Let $\mathcal{D}$ be a category. A *weak factorization system* on $\mathcal{D}$ is a pair $(\mathcal{L}, \mathcal{R})$ of classes of morphisms in $\mathcal{D}$ such that:

(1) (*factorization axiom*) every morphism f in $\mathcal{D}$ can be factored as $f = pi$, where $i \in \mathcal{L}$ and $p \in \mathcal{R}$;

(2) (*weak orthogonality axiom*) $\mathcal{L}^{\pitchfork} = \mathcal{R}$ and $\mathcal{L} = {}^{\pitchfork}\mathcal{R}$.

Remark 7.3. Let $(\mathcal{L}, \mathcal{R})$ be a weak factorization system on a category $\mathcal{D}$. Then both $\mathcal{L}$ and $\mathcal{R}$ are closed under compositions and retracts and contain all isomorphisms. Furthermore, $\mathcal{L}$ is closed under pushouts and coproducts in $\mathcal{D}$ (if they exist), whereas $\mathcal{R}$ is closed under pullbacks and products (if they exist). All these properties follow easily by the characterization of $\mathcal{L}$ and $\mathcal{R}$ in terms of lifting properties.

Lemma 7.4. [Hov99, Lemma 1.1.9] (THE RETRACT ARGUMENT). *Let $\mathcal{D}$ be a category and f a morphism in $\mathcal{D}$. If $f = pi$ is a factorization of f and f has the left lifting property with respect to p, then f is a retract of i. Dually, if f has the right lifting property with respect to i, then f is a retract of p.* □

Definition 7.5. [Lum11, cf. Definition 9] A *pre-model category structure* on a category $\mathcal{D}$ is a triple

$$(\mathcal{W},\ \mathcal{F},\ \mathcal{C})$$

of classes of morphisms in $\mathcal{D}$, called *weak equivalences*, *fibrations* and *cofibrations* respectively, such that:

(1) $\mathcal{W}$ satisfies the 2-out-of-3 property, i.e. given composable maps $f\colon X \to Y$ and $g\colon Y \to Z$, if two among f, g and gf belongs to $\mathcal{W}$, then so does the third;
(2) $(\mathcal{C}, \mathcal{F} \cap \mathcal{W})$ and $(\mathcal{C} \cap \mathcal{W}, \mathcal{F})$ are weak factorization systems in $\mathcal{D}$.

If $(\mathcal{W},\ \mathcal{F},\ \mathcal{C})$ is a pre-model structure on $\mathcal{D}$, then the maps in $\mathcal{F} \cap \mathcal{W}$ are called *trivial fibrations* (or *acyclic fibrations*), whereas the maps in $\mathcal{C} \cap \mathcal{W}$ are called *trivial cofibrations* (or *acyclic cofibrations*). A *pre-model category* is a category $\mathcal{D}$ equipped with a pre-model structure $(\mathcal{W},\ \mathcal{F},\ \mathcal{C})$ on $\mathcal{D}$.

Definition 7.6. Given a pre-model category $(\mathcal{D},\ \mathcal{W},\ \mathcal{F},\ \mathcal{C})$, its *homotopy category* is the localization of $\mathcal{D}$ and $\mathcal{W}$ and is denoted by $\mathrm{Ho}(\mathcal{D})$ (or $\mathrm{Ho}(\mathcal{D},\ \mathcal{W})$ if needed).

Definition 7.7. A *model category* is a pre-model category $(\mathcal{M},\ \mathcal{W},\ \mathcal{F},\ \mathcal{C})$ such that $\mathcal{M}$ is complete and cocomplete and there exist functorial factorizations of each map in $\mathcal{W}$ as a map in $\mathcal{C} \cap \mathcal{W}$ followed by a map in $\mathcal{F}$ and as a map in $\mathcal{C}$ followed by a map in $\mathcal{F} \cap \mathcal{W}$.

Example 7.8. [Hov99, Section 2.4] Let Top be the category of topological spaces and continuous maps among them. There is a model category structure on Top, called the *Quillen model structure* on Top, given as follows:

(i) the weak equivalences are the *weak homotopy equivalences*, i.e. those maps $f\colon X \to Y$ of topological spaces which induce isomorphisms on all homotopy groups (for every choice of the basepoint) and on the set of path components;
(ii) the fibrations are the *Serre fibrations*, i.e. those maps $p\colon E \to X$ of spaces which have the right lifting property with respect to the inclusions $I^n \hookrightarrow I^{n+1}$, for all $n \in \mathbb{N}$ and with I being the unit interval $[0,1]$ ($I^0 := \{0\}$);
(iii) the cofibrations are the maps with the left lifting property with respect to all Serre fibrations which are also weak homotopy equivalences.

Example 7.9. [Hov99, Chapter 3] Let sSet be the category of simplicial sets, that is $\mathrm{sSet} := \mathrm{Set}^{\Delta^{op}}$, where Δ is the simplex category whose objects are the ordinals $[n] = n + 1$, for $n \in \mathbb{N}$. There is a model category structure on sSet, called again the *(Kan-)Quillen model structure* on sSet, given as follows:

(i) the weak equivalences are the morphisms $f\colon X \to Y$ of simplicial sets such that their *geometric realization* $|f|\colon |X| \to |Y|$ is a weak homotopy equivalence;
(ii) the fibrations are the *Kan fibrations*, i.e. those maps $p\colon E \to B$ of simplicial sets which have the right lifting property with respect to every inclusion $\Lambda^k[n] \hookrightarrow \Delta[n]$, for all $n \in \mathbb{N}$ and all $0 \leq k \leq n$. (Recall that $\Delta[n]$ is the representable presheaf $\Delta(-, [n])$ for $[n] \in \Delta$ and $\Lambda^k[n]$ is the $(n,k)-$th *horn*, i.e. the subsimplicial set of $\Delta[n]$ given by the union of all faces except the $k-$th one);
(iii) the cofibrations are the monomorphisms.

REFERENCES

[AKL15] Jeremy Avigad, Krzysztof Kapulkin, and Peter LeFanu Lumsdaine, *Homotopy limits in type theory*, Math. Structures Comput. Sci. **25** (2015), no. 5, 1040–1070, doi:10.1017/S0960129514000498.

[Bro73] Kenneth S. Brown, *Abstract homotopy theory and generalized sheaf cohomology*, Trans. Amer. Math. Soc. **186** (1973), 419–458.

[Gar09] Richard Garner, *Two-dimensional models of type theory*, Math. Structures Comput. Sci. **19** (2009), no. 4, 687–736, doi:10.1017/S0960129509007646.

[GG08] Nicola Gambino and Richard Garner, *The identity type weak factorisation system*, Theoret. Comput. Sci. **409** (2008), no. 1, 94–109, doi:10.1016/j.tcs.2008.08.030.

[Hov99] Mark Hovey, *Model categories*, Mathematical Surveys and Monographs, vol. 63, American Mathematical Society, Providence, RI, 1999.

[Lum11] Peter LeFanu Lumsdaine, *Model structures from higher inductive types*, Preprint (2011).

[Pit00] Andrew M. Pitts, *Categorical logic*, Handbook of logic in computer science, Vol. 5, Handb. Log. Comput. Sci., vol. 5, Oxford Univ. Press, New York, 2000, pp. 39–128.

[Pro13] The Univalent Foundations Program, *Homotopy type theory—univalent foundations of mathematics*, The Univalent Foundations Program, Princeton, NJ; Institute for Advanced Study (IAS), Princeton, NJ, 2013.