

A type theory for synthetic ∞ -categories

Emily Riehl^a and Michael Shulman^b

^aDept. of Mathematics, Johns Hopkins U., 3400 N Charles St., Baltimore, MD 21218

^bDept. of Mathematics, University of San Diego, 5998 Alcalá Park, San Diego, CA 92110

Abstract

We propose foundations for a synthetic theory of $(\infty, 1)$ -categories within homotopy type theory. We axiomatize a directed interval type, then define higher simplices from it and use them to probe the internal categorical structures of arbitrary types. We define *Segal types*, in which binary composites exist uniquely up to homotopy; this automatically ensures composition is coherently associative and unital at all dimensions. We define *Rezk types*, in which the categorical isomorphisms are additionally equivalent to the type-theoretic identities — a “local univalence” condition. And we define *covariant fibrations*, which are type families varying functorially over a Segal type, and prove a “dependent Yoneda lemma” that can be viewed as a directed form of the usual elimination rule for identity types. We conclude by studying homotopically correct adjunctions between Segal types, and showing that for a functor between Rezk types to have an adjoint is a mere proposition.

To make the bookkeeping in such proofs manageable, we use a three-layered type theory with shapes, whose contexts are extended by polytopes within directed cubes, which can be abstracted over using “extension types” that generalize the path-types of cubical type theory. In an appendix, we describe the motivating semantics in the Reedy model structure on bisimplicial sets, in which our Segal and Rezk types correspond to Segal spaces and complete Segal spaces.

Communicated by: Richard Garner.

Received: 22nd June, 2017. Accepted: 27th November, 2017.

MSC: 03G30, 18G55, 55U40.

Keywords: homotopy type theory, ∞ -categories, Segal spaces, Rezk spaces.

1. Introduction

Homotopy type theory [29] is a new subject that augments Martin-Löf constructive dependent type theory with additional rules and axioms enabling it to be used as a formal language for reasoning about homotopy theory. These rules and axioms are motivated by homotopy-theoretic

Email addresses: eriehl@math.jhu.edu (Emily Riehl)

shulman@sandiego.edu (Michael Shulman)

©Emily Riehl and Michael Shulman, 2017, under a Creative Commons Attribution 4.0 International License.

DOI: [10.21136/HS.2017.06](https://doi.org/10.21136/HS.2017.06)

models such as Voevodsky’s simplicial set model [12]. In the latter, the *types* of type theory are interpreted as simplicial sets in the Quillen model structure, which are a presentation of ∞ -groupoids. Thus, homotopy type theory can be viewed as a “synthetic theory of ∞ -groupoids” and a foundational system for higher-categorical mathematics.

Of course, higher category theory is not just about ∞ -groupoids, but also n -categories, $(\infty, 1)$ -categories, (∞, n) -categories, and so on. But a *directed* type theory that could serve as a synthetic theory of such objects has proven somewhat elusive. In particular, one of the advantages of homotopy type theory is that the single simple rule of identity-elimination automatically generates all the higher structure of ∞ -groupoids, whereas (for instance) the 2-dimensional type theory of [16] has to put in the categorical structure by hand, and thereby lacks as much advantage over explicit definitions of categories inside set theory. Moreover, interpreting types directly as (higher) categories runs into various problems, such as the fact that not all maps between categories are exponentiable (so that not all Π -types exist), and that there are numerous different kinds of “fibrations” given the various possible functorialities and dimensions of categories appearing as fibers.

There is no reason in principle to think these problems insurmountable, and many possible solutions have been proposed. However, in this paper we pursue a somewhat indirect route to a synthetic theory of higher categories, which has its own advantages, and may help illuminate some aspects of what an eventual more direct theory might look like. Our approach is based on the following idea, which was also suggested independently by Joyal.

Homotopy type theory admits semantics not only in simplicial sets (hence ∞ -groupoids), but in many other model categories. In particular, as shown in [27], it can be interpreted in the Reedy model structure on *bisimplicial* sets, also called *simplicial spaces*. This model structure, in turn, admits a left Bousfield localization called the *complete Segal space* model structure [22], which presents the homotopy theory and indeed also the category theory [25] of $(\infty, 1)$ -categories. We cannot interpret homotopy type theory (in its usual form) in the complete Segal space model structure directly (due to its lack of right properness among other things), but we can interpret it in the Reedy model structure and identify internally *some* types that correspond to complete Segal spaces. That is, in contrast to ordinary homotopy type theory where the basic objects (types) are exactly the “synthetic ∞ -groupoids”, in our theory the basic objects (types) are something more general, inside of which we identify a class that we regard as “synthetic $(\infty, 1)$ -categories”.

The identification of these “category-like types”, and the study of their properties, depends on adding certain structure to homotopy type theory that is characteristic of the bisimplicial set model. The fundamental such structure is a “directed interval” type, which (thinking categorically) we denote $\mathbb{2}$. The homotopy theoretic analysis of Joyal and Tierney [11] suggests that it is productive to think of bisimplicial sets as having a “spatial” direction and “categorical” direction; simplicial sets can then be embedded in the categorical direction as *discrete* simplicial spaces or in the spatial direction as *constant* simplicial spaces. The semantics of the “directed interval” $\mathbb{2}$ as a bisimplicial set is the simplicial interval Δ^1 , placed in the “categorical” direction rather than the “spatial” direction.¹ As it does in ordinary category theory, the directed interval detects arrows representably: that is, for any type A the function type $\mathbb{2} \rightarrow A$ is the “type of arrows in A ”.

The directed interval $\mathbb{2}$ possesses a lot of useful structure. The *internal* incarnation of this

¹Note that the “spatial” Δ^1 is (weakly) contractible, whereas the “categorical” Δ^1 is not.

structure, which is what is visible in the homotopy type theory of bisimplicial sets, is nicely summarized by saying that it is a *strict interval*: a totally ordered set with distinct bottom and top elements (called 0 and 1). In fact, there is a sense in which it possesses “exactly this structure and no more”: the topos of simplicial sets is the *classifying topos* for such strict interval objects.² If we regard this classifying topos as sitting inside bisimplicial sets in the categorical direction (discrete in the spatial direction), then it is not hard to show that bisimplicial sets similarly present the “classifying $(\infty, 1)$ -topos” of strict intervals; but we will have no need of this.

The strict interval structure on $\mathbb{2}$ (i.e. Δ^1) allows us to define the higher simplices from it internally, and hence the higher categorical structure of types. For instance,

$$\Delta^2 = \{(s, t) : \mathbb{2} \times \mathbb{2} \mid t \leq s\}.$$

We regard a map $\alpha : \Delta^2 \rightarrow A$ as a “commutative triangle” in A witnessing that the composite of $\lambda t. \alpha(t, 0) : \Delta^1 \rightarrow A$ and $\lambda t. \alpha(1, t) : \Delta^1 \rightarrow A$ is $\lambda t. \alpha(t, t) : \Delta^1 \rightarrow A$.

Importantly, for a general type A , two given composable arrows — i.e. two functions $f, g : \mathbb{2} \rightarrow A$ with $f(1) = g(0)$ — may not have any such “composite”, or they may have more than one. If any two composable arrows have a unique composite in the homotopical sense that the type of such composites with their witnesses is contractible, we call A a *Segal type*.

Classically, a *Segal space* is defined as a bisimplicial set X for which *all* the Segal maps $X_n \rightarrow X_1 \times_{X_0} \cdots \times_{X_0} X_1$ are equivalences, thereby saying not only that any two composable arrows have a unique composite, but that any *finite string* of composable arrows has a unique composite. This ensures that composition is associative and unital up to all higher homotopies. Our definition of Segal type appears to speak only about composable *pairs*, but because it is phrased in the internal type theory of bisimplicial sets, semantically it corresponds to asserting not just that the Segal map $X_2 \rightarrow X_1 \times_{X_0} X_1$ is an equivalence of *simplicial sets*, but that the analogous map $X^{\Delta^2} \rightarrow X^{\Delta^1} \times_X X^{\Delta^1}$ is an equivalence of *bisimplicial sets*. Joyal conjectured that this is equivalent to the usual definition of a Segal space; in an appendix we prove this conjecture, justifying our terminology. We can also prove internally in the type theory that composition in a Segal type is automatically associative and so on, so that it behaves just like a category.

Note the strong similarity to how ordinary homotopy type theory functions as a synthetic language for ∞ -groupoids. An explicit ∞ -groupoid is a very complicated structure, but when working “internally” it suffices to equip every type with the single operation of identity-elimination. It then automatically follows, as a meta-theorem, that every type *internally* admits all the structure of an ∞ -groupoid, as shown in [30, 17]; but in practical applications we rarely need more than one or two levels of this structure, and we can just “define it as we go”. Similarly, a Segal space or $(\infty, 1)$ -category is a complicated structure with all higher coherences, but when working “internally” it suffices to assume a single contractibility condition to define a Segal type. We do not prove an analogue of [30, 17] for Segal types, but we conjecture that it should be possible; while in practice we generally seem to only need one or two levels that we can “define as we go”.

If a Segal type satisfies a further condition analogous to Rezk’s “completeness” condition for Segal spaces, we call it a *Rezk type*. These are the ones that semantically model $(\infty, 1)$ -categories. However, for much of the theory it suffices to work with Segal types, which also have an $(\infty, 1)$ -categorical interpretation: they correspond to an $(\infty, 1)$ -category \mathcal{A} equipped with a functor $\mathcal{G} \rightarrow \mathcal{A}$ where \mathcal{G} is an ∞ -groupoid. In [3] this is called a *flagged $(\infty, 1)$ -category*; it can also be thought of as an “ $(\infty, 1)$ -double category” with “connections” and one direction invertible.

²This result was apparently first announced by Joyal at the Isle of Thorns; proofs can be found in [8, 20].

The Rezk types correspond to the flagged $(\infty, 1)$ -categories for which \mathcal{G} is the *core* of \mathcal{A} , the locally full sub- $(\infty, 1)$ -category of invertible morphisms. Note that the need for a “completeness” condition, or equivalently the fact that flagged categories must be defined before unflagged ones, also arises when *defining* categories and higher categories inside homotopy type theory; see [29, Chapter 9] or [1].

The goal of this paper is to develop the basic category theory of Segal and Rezk types. We discuss the behavior of “functors”, which internally are simply functions between such types, and “natural transformations”, which are simply functions $A \times 2 \rightarrow B$. We define what it means for a type family $C : A \rightarrow \mathcal{U}$ to be *covariant* or *contravariant*, and we prove a “dependent Yoneda lemma” that generalizes the usual Yoneda lemma and has the form of a “directed” version of the usual identity-elimination rule.

Many of the theorems are very similar to their versions in ordinary category theory and/or other forms of $(\infty, 1)$ -category theory. In particular, when interpreted in the simplicial spaces model, our synthetic Yoneda lemma provides new proofs of the results that [25, 13, 6, 21] achieve semantically by working with simplicial spaces. But often there is a significant “internalization” benefit, arising from the fact that all type-theoretic functions between Segal types are automatically “functorial” or “natural”. In this sense our theory achieves much of the expected benefit of a “directed homotopy type theory” for studying $(\infty, 1)$ -categories synthetically, with the added advantage that we have the full power of ordinary homotopy type theory to work with (including, for instance, all Π -types) and can draw on all of its results. The presence of non-Segal types, whose category-theoretic meaning is somewhat unclear but which we can ignore whenever we wish, seems a small price to pay.

As a “capstone” application, we study *adjunctions* between Segal and Rezk types, proving the equivalence of a diagrammatic “unit and counit” definition with an “equivalence of homs” definition. As shown in [24], while a complete homotopy-coherent (diagrammatic) adjunction contains infinitely much data, it is uniquely determined by many finite subcollections of data whenever the adjoint relationship exists, such as: a single functor; both functors and the counit; both the unit and counit and a witness of one triangle identity; or witnesses of both triangle identities and a coherence between them (in the last case, no further existence assumptions are required). We show that when transferred to the “equivalence of homs” definition, such subcollections correspond to the finitary coherent definitions of equivalence in homotopy type theory from [29, Chapter 4]. Transferring the “bi-invertibility” definition of equivalence back across this comparison leads to a new way to fully characterize a homotopy coherent adjunction (with no further assumptions): two functors, a unit, and *two* counits, one equipped with a witness that it satisfies one triangle identity and the other equipped with a witness of the other triangle identity.

There is one further technical device we will use, which is of some interest in its own right. In principle, all of the above theory could be developed within ordinary homotopy type theory, simply by axiomatically assuming the type 2 and its strict interval structure. However, we often want to talk about, given two points $x, y : A$, the “type of arrows from x to y ”, i.e. the type of functions $f : 2 \rightarrow A$ such that $f(0) = x$ and $f(1) = y$. If we define this type internally in ordinary homotopy type theory, these latter equalities can only be points of the identity type, so we would have to define

$$\mathrm{hom}_A(x, y) := \sum_{f:2 \rightarrow A} (x = f(0)) \times (f(1) = y).$$

These equalities are then data, which have to be carried around everywhere. This is quite tedious, and the technicalities become nearly insurmountable when we come to define commutative triangles, let alone commutative tetrahedra.

Intuitively, we would like $\text{hom}_A(x, y)$ to be the type of functions $f : \mathbf{2} \rightarrow A$ such that $f(0)$ and $f(1)$ are *strictly*, or *judgmentally*, equal to x and y respectively. Ordinary intensional type theory does not allow us to assert judgmental equalities as data, and the semantic reason for this is that it would not preserve fibrancy: judgmental equality on A is interpreted by the diagonal $A \rightarrow A \times A$, which is not a fibration (unlike the path-object $PA \rightarrow A \times A$, which interprets the identity type).

However, in our motivating model of bisimplicial sets, the “object of functions $f : \mathbf{2} \rightarrow A$ such that $f(0) \equiv x$ and $f(1) \equiv y$ strictly” is fibrant, because the inclusion $\mathbf{2} \rightarrow \mathbf{2}$ is a *cofibration* and the Reedy model structure is cartesian monoidal. The latter ensures that for any cofibration $A \rightarrow B$ and fibration $C \rightarrow D$, the “pullback corner map” or “Leibniz hom”

$$C^B \rightarrow C^A \times_{D^A} D^B$$

is a fibration. Applied to the cofibration $\mathbf{2} \rightarrow \mathbf{2}$ and the fibration $A \rightarrow \mathbf{1}$ we obtain a fibration $A^{\mathbf{2}} \rightarrow A \times A$ representing the desired type family $\text{hom}_A : A \times A \rightarrow \mathcal{U}$.

It is therefore natural to try to “internalize” this argument. There are many possible ways to do this. One “brute force” approach is to use a two-level type theory [31, 2] in which there are both “fibrant types” and “non-fibrant types”, with a non-fibrant “strict equality type” that reifies judgmental equality. We could then define

$$\text{hom}_A(x, y) := \sum_{f : \mathbf{2} \rightarrow A} (x \equiv f(0)) \times (f(1) \equiv y).$$

using strict equalities, and assert axiomatically that it is fibrant, since in general it would not be.

We will use instead a more refined approach that eliminates the need for strict equality and non-fibrant types. We have a judgmental notion of *cofibration*, and a new type former called an *extension type*: if $i : A \rightarrow B$ is a cofibration and $C : B \rightarrow \mathcal{U}$ is a type family over its codomain with a section $d : \prod_{x:A} C(i(x))$ over its domain, then there is a type $\left\langle \prod_{y:B} C(y) \Big|_d^i \right\rangle$ of “dependent functions $f : \prod_{y:B} C(y)$ such that $f(i(x)) \equiv d(x)$ for all $x : A$ ”. This idea is due to Lumsdaine and the second author (unpublished).

We then have to give rules for what counts as a cofibration, in which we have to be careful to respect the semantics: it cannot simply be a map in any context that becomes a cofibration in the semantic slice category, since arbitrary slice categories are no longer cartesian monoidal model categories. However, we need not only $\mathbf{2} \rightarrow \mathbf{2}$ to be a cofibration, but also the inclusion of the boundary of any simplex $\partial\Delta^n \rightarrow \Delta^n$, and we would like these to be constructible in a sensible and uniform way rather than axiomatically asserted. One approach would be to keep the non-fibrant types with a notion of “strict pushout”, and rules that cofibrations are closed under operations such as the “pushout product” or “pushout join”.

We instead choose to keep all types fibrant (and hence all proofs more clearly homotopy-invariant), introducing rather a syntax for specifying cofibrations entirely separately from the rest of the type theory. Pleasingly, this separate syntax is exactly the coherent theory of a strict interval. We have a judgmental notion of *shape*, representing the polytopes embedded in directed cubes that can be constructed in the theory of a strict interval, and we take the cofibrations to

be the “inclusions of sub-shapes”. For instance, the boundary of Δ^2 is the shape

$$\partial\Delta^2 := \{\langle s, t \rangle : 2 \times 2 \mid (t \leq s) \wedge (t \equiv 0 \vee s \equiv t \vee s \equiv 1)\}$$

This choice also makes the setup more flexible, since in principle any other suitable theory could be used instead. For instance, using Joyal’s theory of disks [10] would presumably yield a type theory in which to study (∞, n) -categories in the style of [23]. In an appendix we sketch how our setup should be interpretable semantically in bisimplicial sets.

Remark 1.1. Formally, our theory is very similar to the recent “cubical type theories” studied by [5] and others, whose basic setup can also be regarded as an instance of ours, using the theory of a de Morgan algebra. The most substantial difference is that our interval 2 describes extra structure in an “orthogonal” direction to the native “homotopy theory” of homotopy type theory, whereas the cubical interval is rather a different way of describing that exact same native homotopy theory. This is why cubical type theory also includes the cubical Kan operations as rules of type theory; the closest analogue of this in our theory is the category structure of a Segal type induced by the contractibility of its composition spaces.

We introduce our basic type theory with shapes in §2, and specialize to the simplicial type theory using the strict interval in §3. In §4 we prove some basic results about extension types and how they commute with each other and with the other type constructors.

Then in §5 we give the basic definition of a Segal type and study its structure as a sort of “category”, while in §6 we study the corresponding behavior of “functors” and “natural transformations”. Section 7 is devoted to a special kind of Segal type that we call “discrete”; semantically these correspond to homotopically constant simplicial spaces; if Segal and Rezk types are the “categories”, discrete types are the “groupoids”. Then in §8 we study covariant and contravariant type families, which are families of discrete types that vary functorially over a Segal type; these are the synthetic analogue of covariant and contravariant fibrations or presheaves. In particular, they satisfy the Yoneda lemma, as we show in §9.

In §10 we define Rezk types, which are Segal types satisfying a “completeness” or “univalence” condition identifying the type-theoretic identity type with the categorical isomorphisms. And in §11 we study homotopy coherent adjunctions between Segal and Rezk types.

Finally, in Appendix A we briefly discuss the motivating semantics in bisimplicial sets and other “model categories with shapes”, and show that our Segal and Rezk types correspond to Segal spaces and complete Segal spaces. The analogous correspondence for covariant fibrations follows from recent work of [25, 13, 6, 21].

2. Type theory with shapes

Our type theory has three layers. The first two are basically ordinary coherent first-order logic, in which we express the theory of a strict interval; the third layer is then a homotopy type theory over the first two. For clarity and generality, in this section we describe only the formal apparatus of the type theory; in §3 we will then add to it the axioms of a strict interval that we will use in the rest of the paper.

2.1 Cubes, topes, shapes, and types The first layer is a simple intuitionistic type theory with finite product types and nothing else. We call the types in this layer **cubes**; in our theory

$$\begin{array}{c}
 \frac{}{\mathbf{1} \text{ cube}} \qquad \frac{I \text{ cube} \quad J \text{ cube}}{I \times J \text{ cube}} \qquad \frac{(t : I) \in \Xi}{\Xi \vdash t : I} \qquad \frac{}{\Xi \vdash \star : \mathbf{1}} \\
 \\
 \frac{\Xi \vdash s : I \quad \Xi \vdash t : J}{\Xi \vdash \langle s, t \rangle : I \times J} \qquad \frac{\Xi \vdash t : I \times J}{\Xi \vdash \pi_1(t) : I} \qquad \frac{\Xi \vdash t : I \times J}{\Xi \vdash \pi_2(t) : J}
 \end{array}$$

Figure 1: The cube layer

they will be finite powers of the one “generating cube” $\mathbf{2}$. The formal rules for the cube layer are shown in Figure 1; here Ξ is a context of variables belonging to cubes.

The second layer is an intuitionistic logic over the first. We refer to its types as **topes**, thinking of them as polytopes embedded in a cube (namely, the “cube context” Ξ). Topes admit operations of finite conjunction and disjunction, but not negation, implication, or either quantifier.³ There is also a basic “equality tope”, which we write with the symbol \equiv , since it will be visible to the third layer as a “strict” or “judgmental” equality. (In the theory of a strict interval introduced in §3.1, there will also be an *inequality* tope.)

The formal rules of the tope layer are shown in Figure 2; here Φ is a list of topes. Note that we include the β and η rules for finite product cubes as introductions for equality topes. We state all the rules in “natural deduction style”, ensuring the admissibility of the usual structural rules like weakening, contraction, substitution, and cut. For instance, here are the substitution rules:

$$\frac{\Xi \vdash t : I \quad \Xi, x : I \vdash \phi \text{ tope}}{\Xi \vdash \phi[t/x] \text{ tope}} \qquad \frac{\Xi \vdash t : I \quad \Xi, x : I \mid \Phi \vdash \psi}{\Xi \mid \Phi[t/x] \vdash \psi[t/x]}$$

and here is the cut rule for topes:

$$\frac{\Xi \mid \Phi \vdash \psi \quad \Xi \mid \Phi, \psi \vdash \chi}{\Xi \mid \Phi \vdash \chi}$$

By a **shape** we will mean a cube together with a tope in the corresponding singleton context. We could formalize this with a judgment and introduction rule such as the following:

$$\frac{I \text{ cube} \quad t : I \vdash \phi \text{ tope}}{\{t : I \mid \phi\} \text{ shape}} \tag{2.1}$$

The most important shapes for us will be the n -simplices and their boundaries and partial boundaries (such as horns).

Finally, the third layer is an ordinary intensional dependent type theory in which every judgment has additional contexts of cubes and topes. All the usual type formers and rules leave these cube and tope contexts unchanged. As in [29], we include Σ -types, Π -types with judgmental η -conversion, coproduct types, identity types $x : A, y : A \vdash x = y$ type, a universe \mathcal{U} (but see Remark 2.5), and so on. We assume function extensionality as in [29, §2.9], but we will not need any higher inductive types, nor the univalence axiom (although we expect that it, or at least directed analogues of it, will become important as the theory is developed further).

³We could probably include the existential quantifier to obtain a full “coherent logic”, and possibly even go beyond this, but for the theory of a strict interval we only need conjunction and disjunction.

$$\begin{array}{c}
\frac{\phi \in \Phi}{\Xi | \Phi \vdash \phi} \quad \frac{}{\Xi \vdash \top \text{ tope}} \quad \frac{}{\Xi | \Phi \vdash \top} \quad \frac{}{\Xi \vdash \perp \text{ tope}} \quad \frac{\Xi | \Phi \vdash \perp}{\Xi | \Phi \vdash \psi} \\
\frac{\Xi \vdash \phi \text{ tope} \quad \Xi \vdash \psi \text{ tope}}{\Xi \vdash (\phi \wedge \psi) \text{ tope}} \quad \frac{\Xi | \Phi \vdash \phi \quad \Xi | \Phi \vdash \psi}{\Xi | \Phi \vdash \phi \wedge \psi} \quad \frac{\Xi | \Phi \vdash \phi \wedge \psi}{\Xi | \Phi \vdash \phi} \quad \frac{\Xi | \Phi \vdash \phi \wedge \psi}{\Xi | \Phi \vdash \psi} \\
\frac{\Xi \vdash \phi \text{ tope} \quad \Xi \vdash \psi \text{ tope}}{\Xi \vdash (\phi \vee \psi) \text{ tope}} \quad \frac{\Xi | \Phi \vdash \phi}{\Xi | \Phi \vdash \phi \vee \psi} \quad \frac{\Xi | \Phi \vdash \psi}{\Xi | \Phi \vdash \phi \vee \psi} \\
\frac{\Xi | \Phi, \phi \vdash \chi \quad \Xi | \Phi, \psi \vdash \chi}{\Xi | \Phi \vdash \chi} \quad \frac{\Xi | \Phi \vdash \phi \vee \psi}{\Xi \vdash s : I \quad \Xi \vdash t : I} \quad \frac{\Xi \vdash s : I}{\Xi | \Phi \vdash (s \equiv t) \text{ tope}} \quad \frac{\Xi \vdash s : I}{\Xi | \Phi \vdash (s \equiv s)} \\
\frac{\Xi | \Phi \vdash (s \equiv t)}{\Xi | \Phi \vdash (t \equiv s)} \quad \frac{\Xi | \Phi \vdash (s \equiv t) \quad \Xi | \Phi \vdash (t \equiv v)}{\Xi | \Phi \vdash (s \equiv v)} \\
\frac{\Xi | \Phi \vdash (s \equiv t) \quad \Xi, x : I \vdash \psi \text{ tope} \quad \Xi | \Phi \vdash \psi[s/x]}{\Xi | \Phi \vdash \psi[t/x]} \quad \frac{\Xi \vdash t : \mathbf{1}}{\Xi | \Phi \vdash t \equiv \star} \\
\frac{\Xi \vdash s : I \quad \Xi \vdash t : J}{\Xi | \Phi \vdash \pi_1(\langle s, t \rangle) \equiv s} \quad \frac{\Xi \vdash s : I \quad \Xi \vdash t : J}{\Xi | \Phi \vdash \pi_2(\langle s, t \rangle) \equiv t} \quad \frac{\Xi \vdash t : I \times J}{\Xi | \Phi \vdash t \equiv \langle \pi_1(t), \pi_2(t) \rangle}
\end{array}$$

Figure 2: The tope layer

$$\begin{array}{c}
\frac{\Xi \mid \Phi \vdash \perp}{\Xi \mid \Phi \mid \Gamma \vdash \text{rec}_{\perp} : A} \qquad \frac{\Xi \mid \Phi \vdash \perp \quad \Xi \mid \Phi \mid \Gamma \vdash a : A}{\Xi \mid \Phi \mid \Gamma \vdash a \equiv \text{rec}_{\perp}} \\
\\
\frac{\Xi \mid \Phi \vdash \phi \vee \psi \quad \Xi \mid \Phi \mid \Gamma \vdash A \text{ type} \quad \Xi \mid \Phi, \phi \mid \Gamma \vdash a_{\phi} : A \quad \Xi \mid \Phi, \psi \mid \Gamma \vdash a_{\psi} : A \quad \Xi \mid \Phi, \phi \wedge \psi \mid \Gamma \vdash a_{\phi} \equiv a_{\psi}}{\Xi \mid \Phi \mid \Gamma \vdash \text{rec}_{\vee}^{\phi, \psi}(a_{\phi}, a_{\psi}) : A} \\
\\
\frac{\Xi \mid \Phi \vdash \phi \vee \psi \quad \Xi \mid \Phi \mid \Gamma \vdash A \text{ type} \quad \Xi \mid \Phi, \phi \mid \Gamma \vdash a_{\phi} : A \quad \Xi \mid \Phi, \psi \mid \Gamma \vdash a_{\psi} : A \quad \Xi \mid \Phi, \phi \wedge \psi \mid \Gamma \vdash a_{\phi} \equiv a_{\psi}}{\Xi \mid \Phi, \phi \mid \Gamma \vdash \text{rec}_{\vee}^{\phi, \psi}(a_{\phi}, a_{\psi}) \equiv a_{\phi}} \\
\\
\frac{\Xi \mid \Phi \vdash \phi \vee \psi \quad \Xi \mid \Phi \mid \Gamma \vdash A \text{ type} \quad \Xi \mid \Phi, \phi \mid \Gamma \vdash a_{\phi} : A \quad \Xi \mid \Phi, \psi \mid \Gamma \vdash a_{\psi} : A \quad \Xi \mid \Phi, \phi \wedge \psi \mid \Gamma \vdash a_{\phi} \equiv a_{\psi}}{\Xi \mid \Phi, \psi \mid \Gamma \vdash \text{rec}_{\vee}^{\phi, \psi}(a_{\phi}, a_{\psi}) \equiv a_{\psi}} \\
\\
\frac{\Xi \mid \Phi \vdash \phi \vee \psi \quad \Xi \mid \Phi \mid \Gamma \vdash a : A}{\Xi \mid \Phi \mid \Gamma \vdash a \equiv \text{rec}_{\vee}^{\phi, \psi}(a, a)}
\end{array}$$

Figure 3: Type elimination for tope disjunction

In addition, we have various rules that relate the first two layers to the third. Firstly, we state all the rules in such a way that the following substitution/cut rules are admissible:

$$\frac{\Xi \vdash t : I \quad \Xi, x : I \mid \Phi \mid \Gamma \vdash a : A}{\Xi \mid \Phi[t/x] \mid \Gamma[t/x] \vdash a[t/x] : A[t/x]} \qquad \frac{\Xi \mid \Phi \vdash \psi \quad \Xi \mid \Phi, \psi \mid \Gamma \vdash a : A}{\Xi \mid \Phi \mid \Gamma \vdash a : A}$$

along with the obvious rules like weakening and contraction for the cube and tope contexts.

Secondly, we have rules ensuring that the type theory respects the “tope logic” in a strict judgmental way. The appropriate sort of respect for \top and \wedge is already ensured by the cut and weakening rules. For instance, we have the following derivations for \wedge :

$$\begin{array}{c}
\frac{\Xi \mid \Phi, \phi, \psi \mid \Gamma \vdash a : A}{\Xi \mid \Phi, \phi \wedge \psi, \phi, \psi \mid \Gamma \vdash a : A} \text{ WEAK} \\
\frac{\Xi \mid \Phi, \phi \wedge \psi, \phi, \psi \mid \Gamma \vdash a : A}{\Xi \mid \Phi, \phi \wedge \psi, \phi \mid \Gamma \vdash a : A} \text{ CUT} \\
\frac{\Xi \mid \Phi, \phi \wedge \psi, \phi \mid \Gamma \vdash a : A}{\Xi \mid \Phi, \phi \wedge \psi \mid \Gamma \vdash a : A} \text{ CUT}
\end{array}
\qquad
\begin{array}{c}
\frac{\Xi \mid \Phi, \phi \wedge \psi \mid \Gamma \vdash a : A}{\Xi \mid \Phi, \phi, \phi \wedge \psi \mid \Gamma \vdash a : A} \text{ WEAK} \\
\frac{\Xi \mid \Phi, \phi, \phi \wedge \psi \mid \Gamma \vdash a : A}{\Xi \mid \Phi, \phi, \psi, \phi \wedge \psi \mid \Gamma \vdash a : A} \text{ WEAK} \\
\frac{\Xi \mid \Phi, \phi, \psi, \phi \wedge \psi \mid \Gamma \vdash a : A}{\Xi \mid \Phi, \phi, \psi \mid \Gamma \vdash a : A} \text{ CUT}
\end{array}$$

But in the case of \perp and \vee , we have to assert elimination and computation rules, as shown in Figure 3. Note that the rules for \vee say that $\phi \vee \psi$ is a (strict) *pushout* of ϕ and ψ under $\phi \wedge \psi$, as is always the case in a coherent category.

We also require the following compatibility rule, saying that tope equality behaves like judgmental equality:

$$\frac{\Xi \mid \Phi \vdash (s \equiv t) \quad \Xi, x : I \mid \Phi \mid \Gamma \vdash a : A}{\Xi \mid \Phi \mid \Gamma[s/x] \vdash a[s/x] \equiv a[t/x]} \quad (2.2)$$

Note that in the premise, \equiv refers to the equality tope in the second layer, while in the conclusion it refers to the judgmental equality of the third layer. Also, inductively we have $\Gamma[s/x] \equiv \Gamma[t/x]$, so both terms $a[s/x]$ and $a[t/x]$ in the conclusion are well-typed in the same context.

$$\begin{array}{c}
\frac{\begin{array}{c} \{t : I \mid \phi\} \text{ shape} \quad \{t : I \mid \psi\} \text{ shape} \\ t : I \mid \phi \vdash \psi \quad \Xi \mid \Phi \vdash \Gamma \text{ ctx} \quad \Xi, t : I \mid \Phi, \psi \mid \Gamma \vdash A \text{ type} \quad \Xi, t : I \mid \Phi, \phi \mid \Gamma \vdash a : A \end{array}}{\Xi \mid \Phi \mid \Gamma \vdash \left\langle \prod_{t:I \mid \psi} A \Big|_a^\phi \right\rangle \text{ type}} \\
\\
\frac{\begin{array}{c} \{t : I \mid \phi\} \text{ shape} \quad \{t : I \mid \psi\} \text{ shape} \\ t : I \mid \phi \vdash \psi \quad \Xi \mid \Phi \vdash \Gamma \text{ ctx} \quad \Xi, t : I \mid \Phi, \psi \mid \Gamma \vdash A \text{ type} \quad \Xi, t : I \mid \Phi, \phi \mid \Gamma \vdash a : A \\ \Xi, t : I \mid \Phi, \psi \mid \Gamma \vdash b : A \quad \Xi, t : I \mid \Phi, \phi \mid \Gamma \vdash b \equiv a \end{array}}{\Xi \mid \Phi \mid \Gamma \vdash \lambda t^{I \mid \psi}. b : \left\langle \prod_{t:I \mid \psi} A \Big|_a^\phi \right\rangle} \\
\\
\frac{\begin{array}{c} \{t : I \mid \phi\} \text{ shape} \quad \{t : I \mid \psi\} \text{ shape} \quad t : I \mid \phi \vdash \psi \\ \Xi \mid \Phi \mid \Gamma \vdash f : \left\langle \prod_{t:I \mid \psi} A \Big|_a^\phi \right\rangle \quad \Xi \vdash s : I \quad \Xi \mid \Phi \vdash \psi[s/t] \end{array}}{\Xi \mid \Phi \mid \Gamma \vdash f(s) : A} \\
\\
\frac{\begin{array}{c} \{t : I \mid \phi\} \text{ shape} \quad \{t : I \mid \psi\} \text{ shape} \quad t : I \mid \phi \vdash \psi \\ \Xi \mid \Phi \mid \Gamma \vdash f : \left\langle \prod_{t:I \mid \psi} A \Big|_a^\phi \right\rangle \quad \Xi \vdash s : I \quad \Xi \mid \Phi \vdash \phi[s/t] \end{array}}{\Xi \mid \Phi \mid \Gamma \vdash f(s) \equiv a[s/t]} \\
\\
\frac{\begin{array}{c} \{t : I \mid \phi\} \text{ shape} \quad \{t : I \mid \psi\} \text{ shape} \\ t : I \mid \phi \vdash \psi \quad \Xi \mid \Phi \vdash \Gamma \text{ ctx} \quad \Xi, t : I \mid \Phi, \psi \mid \Gamma \vdash A \text{ type} \quad \Xi, t : I \mid \Phi, \phi \mid \Gamma \vdash a : A \\ \Xi, t : I \mid \Phi, \psi \mid \Gamma \vdash b : A \quad \Xi, t : I \mid \Phi, \phi \mid \Gamma \vdash b \equiv a \quad \Xi \vdash s : I \quad \Xi \mid \Phi \vdash \psi[s/t] \end{array}}{\Xi \mid \Phi \mid \Gamma \vdash (\lambda t^{I \mid \psi}. b)(s) \equiv b[s/t]} \\
\\
\frac{\begin{array}{c} \{t : I \mid \phi\} \text{ shape} \quad \{t : I \mid \psi\} \text{ shape} \quad t : I \mid \phi \vdash \psi \\ \Xi \mid \Phi \mid \Gamma \vdash f : \left\langle \prod_{t:I \mid \psi} A \Big|_a^\phi \right\rangle \end{array}}{\Xi \mid \Phi \mid \Gamma \vdash f \equiv \lambda t^{I \mid \psi}. f(t)}
\end{array}$$

Figure 4: Extension types

2.2 Extension types along cofibrations Finally, we come to the reason for introducing this whole three-layer theory: extension types along cofibrations. As our notion of “cofibration” we use a *shape inclusion*, i.e. a pair of shapes $\{t : I \mid \phi\}$ and $\{t : I \mid \psi\}$ in the same cube such that $t : I \mid \phi \vdash \psi$. We will sometimes abbreviate this as $\{t : I \mid \phi\} \subseteq \{t : I \mid \psi\}$.

The rules for extension types are shown in Figure 4. In the formation rule, the judgment $\Xi \mid \Phi \vdash \Gamma \text{ ctx}$ means that Γ is a well-formed context of types relative to $\Xi \mid \Phi$. The point is that Γ is not allowed to depend on t or ψ , and (implicitly) that Φ is also not allowed to depend on t . The type A , however, is allowed to depend on t and ψ , i.e. we allow “dependent extensions”. The rest of the rules say essentially that an extension type behaves like an ordinary dependent function type, with β and η rules, except that all its elements act like the supplied section $a : A$ whenever ϕ holds.

As with ordinary dependent function types, if the codomain type A does not actually depend on the domain shape $\{t : I \mid \psi\}$, instead of $\left\langle \prod_{t:I \mid \psi} A \Big|_a^\phi \right\rangle$ we write $\left\langle \{t : I \mid \psi\} \rightarrow A \Big|_a^\phi \right\rangle$.

A different special case is when ϕ is \perp . Then the section a might as well be rec_\perp , while all the

required judgmental equalities also hold automatically by rec_\perp , so the extension type behaves just like an ordinary (possibly dependent) function type whose domain is a shape and whose codomain is a type. Thus, we omit the angle brackets, writing

$$\left(\prod_{t:I|\psi} A\right) := \left\langle \prod_{t:I|\psi} A \Big|_{\text{rec}_\perp}^\perp \right\rangle \quad (2.3)$$

$$\left(\{t : I \mid \psi\} \rightarrow A\right) := \left\langle \{t : I \mid \psi\} \rightarrow A \Big|_{\text{rec}_\perp}^\perp \right\rangle. \quad (2.4)$$

Having just introduced extension types and their notation, we now proceed to introduce an abuse of that notation. The rules in Figure 4 are written in the usual formal type-theoretic way, with the dependent type A , tope ϕ , and term $a : A$ being expression metavariables containing the variable $t : I$. Note that the variable t is bound in all three, i.e. its binding in $\prod_{t:I|\psi}$ scopes over the rest of the expression.

However, once we *have* extension types, and when writing informally and internally to type theory (as we will do for most of the paper), it is more readable and natural to regard A as a function into the universe and a as a dependent function

$$A : \{t : I \mid \psi\} \rightarrow \mathcal{U} \qquad a : \prod_{t:I|\phi} A(t).$$

(The types of A and a here are actually also extension types, with \perp implicit as noted above.) It is then natural to write the extension type as

$$\left\langle \prod_{t:I|\psi} A(t) \Big|_a^\phi \right\rangle \quad \text{or} \quad \left\langle \prod_{t:I|\psi} A(t) \Big|_{\lambda t. a(t)}^\phi \right\rangle.$$

Once we introduce notations for important shapes, such as the simplices Δ^n (see §3.2), it will be natural to also use these in place of the tope ϕ :

$$\left\langle \prod_{t:I|\psi} A(t) \Big|_a^{\{t:I|\phi\}} \right\rangle \quad \text{or} \quad \left\langle \prod_{t:I|\psi} A(t) \Big|_{\lambda t. a(t)}^{\{t:I|\phi\}} \right\rangle.$$

Remark 2.5. This notation does technically require a universe type, to be the codomain of A . Our primary motivating model of bisimplicial sets does have a universe by [27], but as we will see in Appendix A.2 there are other interesting models where universes are not known to exist. However, our use of universes in this paper will be only for notational convenience; all the results could equally well be formulated in a type theory without universes.⁴

3. Simplicial type theory

This completes the specification of our general type theory with shapes and extension types. As a special case, we now formulate the theory that we will work in for the rest of the paper, in which the cube and tope layers form the coherent theory of a strict interval. The simplices are then defined as particular shapes in this coherent theory. In §5 we will define hom types by using these simplex shapes as “probes”.

3.1 The strict interval To define the coherent theory of the strict interval we begin with the axiomatic cubes and terms:

$$\underline{\quad\quad\quad} \quad \underline{\quad\quad\quad} \quad \underline{\quad\quad\quad}$$

$$2 \text{ cube} \qquad 0 : 2 \qquad 1 : 2$$

⁴For further development of the theory of synthetic $(\infty, 1)$ -categories, we expect various universes to be necessary though.

and an axiomatic inequality tope

$$x : \mathbb{2}, y : \mathbb{2} \vdash (x \leq y) \text{ tope}$$

together with the following strict interval axioms:

$$\begin{aligned} & x : \mathbb{2} \mid \cdot \vdash (x \leq x) \\ x : \mathbb{2}, y : \mathbb{2}, z : \mathbb{2} \mid (x \leq y), (y \leq z) & \vdash (x \leq z) \\ x : \mathbb{2}, y : \mathbb{2} \mid (x \leq y), (y \leq x) & \vdash (x \equiv y) \\ x : \mathbb{2}, y : \mathbb{2} \mid \cdot \vdash (x \leq y) \vee (y \leq x) & \\ x : \mathbb{2} \mid \cdot \vdash (0 \leq x) & \\ x : \mathbb{2} \mid \cdot \vdash (x \leq 1) & \\ \cdot \mid (0 \equiv 1) \vdash \perp & \end{aligned}$$

(Technically, to maintain admissibility of cut, such axioms should be formulated as inference rules, but this version is much more readable.)

Remark 3.1. Note that this theory has a “duality” involution obtained by interchanging 0 with 1 and reversing the order of \leq . Since the rules of our type theory are independent of the particular tope theory, it follows that our entire three-layer type theory has a duality involution. This is a *syntactic* and “meta-theoretic” involution, transforming every proof into a dual proof; but it also corresponds semantically to the categorical duality involution on bisimplicial sets obtained by precomposing, in the “categorical” but not also in the “spatial” direction, with the functor $(-)^{\circ} : \Delta \rightarrow \Delta$ that reverses the direction on each ordinal. However, there can also be other models that lack such an involution, since it is not internalized in the syntax.

Remark 3.2. The cubical type theory of [5] can (at least approximately) be regarded as a different instance of our type theory with shapes, together with added “Kan operations”. Instead of the coherent theory of a strict interval, [5] uses the coherent theory of a nondegenerate de Morgan algebra, having the following axiomatic cubes and terms:

$$\begin{array}{ccc} \mathbb{1} \text{ cube} & 0 : \mathbb{1} & 1 : \mathbb{1} \\ t : \mathbb{1}, s : \mathbb{1} \vdash t \mathbf{V} s : \mathbb{1} & t : \mathbb{1}, s : \mathbb{1} \vdash t \mathbf{\wedge} s : \mathbb{1} & t : \mathbb{1} \vdash \neg t : \mathbb{1}. \end{array}$$

Note that here $\mathbf{\wedge}$ and \mathbf{V} are cube term constructors denoting lattice operations, not to be confused with the logical conjunction and disjunction \wedge and \vee that act on topes. They are subject to the

axioms of a distributive lattice:

$$\begin{aligned}
 t : \mathbb{1} \mid \cdot \vdash t \vee t &\equiv t \\
 t : \mathbb{1}, s : \mathbb{1} \mid \cdot \vdash t \vee s &\equiv s \vee t \\
 t : \mathbb{1}, s : \mathbb{1}, u : \mathbb{1} \mid \cdot \vdash t \vee (s \vee u) &\equiv (t \vee s) \vee u \\
 t : \mathbb{1} \mid \cdot \vdash t \vee 0 &\equiv t \\
 t : \mathbb{1} \mid \cdot \vdash t \wedge t &\equiv t \\
 t : \mathbb{1}, s : \mathbb{1} \mid \cdot \vdash t \wedge s &\equiv s \wedge t \\
 t : \mathbb{1}, s : \mathbb{1}, u : \mathbb{1} \mid \cdot \vdash t \wedge (s \wedge u) &\equiv (t \wedge s) \wedge u \\
 t : \mathbb{1} \mid \cdot \vdash t \wedge 1 &\equiv t \\
 t : \mathbb{1}, s : \mathbb{1} \mid \cdot \vdash t \vee (t \wedge s) &\equiv t \\
 t : \mathbb{1}, s : \mathbb{1} \mid \cdot \vdash t \wedge (t \vee s) &\equiv t \\
 t : \mathbb{1}, s : \mathbb{1}, u : \mathbb{1} \mid \cdot \vdash t \vee (s \wedge u) &\equiv (t \wedge s) \vee (t \vee u)
 \end{aligned}$$

plus those of a de Morgan algebra:

$$\begin{aligned}
 t : \mathbb{1}, s : \mathbb{1} \mid \cdot \vdash \neg(t \wedge s) &\equiv \neg t \vee \neg s \\
 t : \mathbb{1} \mid \cdot \vdash \neg\neg t &\equiv t
 \end{aligned}$$

and finally distinctness of the top and bottom elements:

$$\cdot \mid (0 \equiv 1) \vdash \perp.$$

The cubical path-type $\mathbf{Path}_A(x, y)$ is the extension type $\left\langle \prod_{t:\mathbb{1}} A \Big|_{\text{rec}_v(x,y)}^{t \equiv 0 \vee t \equiv 1} \right\rangle$ analogous to our hom-types (see §5.1), and the “face lattice” \mathbb{F} of [5] corresponds to the coherent logic of topes, while the “systems” of [5] correspond to the rules in Figure 3. The de Morgan negation \neg on $\mathbb{1}$ yields a “path reversal” operation, while the minima and maxima operations yield path operations constructing “connection squares” such as

$$\begin{array}{ccc}
 x & \xrightarrow{f} & y \\
 f \downarrow & \searrow f & \cdot \\
 y & \xrightarrow{\quad} & y
 \end{array}
 \quad \text{and} \quad
 \begin{array}{ccc}
 x & \xrightarrow{\quad} & x \\
 \cdot & \searrow f & \downarrow f \\
 x & \xrightarrow{f} & y
 \end{array}$$

The composition and Kan operations, however, are something extra in cubical type theory without any analogue in our general type theory with shapes; they force the cubical path-types to represent the “internal” homotopy theory of types, rather than moving in an “orthogonal” direction like our directed hom-types will.

Of course, in our theory we do not want a negation, since it is explicitly supposed to be *directed*. We do not include binary minima and maxima explicitly, but we could do so without changing the coherent theory, since a total order always has minima and maxima. However, as we will see, this same argument enables us to construct connection squares using rec_v , so there is no need for minimum and maximum operations.

There are many possible variations of “cubical type theory”, corresponding to variations in the coherent theory chosen (for instance, leaving out negation and/or maxima and minima). There are also many possible variations of our directed type theory; for instance, the theory of

discs from [10] should yield synthetic theories of (∞, n) - or (∞, ∞) -categories, corresponding semantically to the Θ -spaces of [23]. Moreover, we can combine theories: for instance, with a strict interval 2 and an unrelated de Morgan algebra $\mathbb{1}$, we could obtain a “directed cubical type theory” in which the intrinsic homotopy theory is cubical but there is an additional orthogonal directed structure. Our type theory with shapes supplies a general context in which to investigate a large class of such theories.

3.2 Simplices and their subshapes The interval type 2 allows us to define the **simplices** as the following shapes:⁵

$$\Delta^n := \{ \langle t_1, \dots, t_n \rangle : 2^n \mid t_n \leq \dots \leq t_1 \}$$

We note that this is a meta-theoretic definition, as there is internally no “natural numbers” that we can use to parametrize a “family of shapes” (there is a natural numbers *type*, but shapes are not allowed to depend on types). However, we will only need it for very small concrete values of n , in which case the meaning is clear:

$$\begin{aligned} \Delta^0 &:= \{ t : \mathbf{1} \mid \top \} \\ \Delta^1 &:= \{ t : 2 \mid \top \} \\ \Delta^2 &:= \{ \langle t_1, t_2 \rangle : 2 \times 2 \mid t_2 \leq t_1 \} \\ \Delta^3 &:= \{ \langle t_1, t_2, t_3 \rangle : 2 \times 2 \times 2 \mid t_3 \leq t_2 \leq t_1 \} \end{aligned}$$

Moreover, as we shall discover in §3.3, the higher dimensional simplices may be inductively defined from these lower dimensional ones by means of the join operation (though, again, the induction is external to the type theory).

Remark 3.3. The perhaps-surprising reversal of order in the coordinates is chosen so that t_i parametrizes the i^{th} arrow in the spine of a simplex. For instance, in a 3-simplex $f : \Delta^3 \rightarrow A$ with the following boundary:



we have

$$\begin{aligned} f_1(t) &\equiv f(t, 0, 0) \\ f_2(t) &\equiv f(1, t, 0) \\ f_3(t) &\equiv f(1, 1, t). \end{aligned}$$

The other three 1-simplices are given by

$$\begin{aligned} f_{12}(t) &\equiv f(t, t, 0) \\ f_{23}(t) &\equiv f(1, t, t) \\ f_{123}(t) &\equiv f(t, t, t). \end{aligned}$$

⁵Formally, this should really be something like

$$\{ t : (\dots((2 \times 2) \times 2) \dots) \mid \pi_2(t) \leq \pi_2(\pi_1(t)) \leq \dots \leq (\pi_1)^n(t) \},$$

but no problems will arise from this sort of abuse of notation.

The other face and degeneracy operations between simplices can be defined in analogous ways. For instance, the four 2-simplex faces of a 3-simplex are obtained by requiring $0 \equiv t_3$, $t_3 \equiv t_2$, $t_2 \equiv t_1$, and $t_1 \equiv 1$ respectively. These yield operations on extension types:

$$\begin{aligned} \lambda f. \lambda \langle t_1, t_2 \rangle. f \langle t_1, t_2, 0 \rangle &: (\Delta^3 \rightarrow A) \rightarrow (\Delta^2 \rightarrow A) \\ \lambda f. \lambda \langle t_1, t_2 \rangle. f \langle t_1, t_2, t_2 \rangle &: (\Delta^3 \rightarrow A) \rightarrow (\Delta^2 \rightarrow A) \\ \lambda f. \lambda \langle t_1, t_2 \rangle. f \langle t_1, t_1, t_2 \rangle &: (\Delta^3 \rightarrow A) \rightarrow (\Delta^2 \rightarrow A) \\ \lambda f. \lambda \langle t_1, t_2 \rangle. f \langle 1, t_1, t_2 \rangle &: (\Delta^3 \rightarrow A) \rightarrow (\Delta^2 \rightarrow A). \end{aligned}$$

Similarly, the two degenerate 2-simplices associated to a 1-simplex are given by ignoring one variable:

$$\begin{aligned} \lambda f. \lambda \langle t_1, t_2 \rangle. f(t_1) &: (\Delta^1 \rightarrow A) \rightarrow (\Delta^2 \rightarrow A) \\ \lambda f. \lambda \langle t_1, t_2 \rangle. f(t_2) &: (\Delta^1 \rightarrow A) \rightarrow (\Delta^2 \rightarrow A) \end{aligned}$$

and so on.

We will also use various sub-shapes of the simplices, particularly their **boundaries**:

$$\begin{aligned} \partial\Delta^1 &:= \{t : 2 \mid (0 \equiv t) \vee (t \equiv 1)\} \\ \partial\Delta^2 &:= \{\langle t_1, t_2 \rangle : 2 \times 2 \mid (0 \equiv t_2 \leq t_1) \vee (t_2 \equiv t_1) \vee (t_2 \leq t_1 \equiv 1)\} \end{aligned}$$

The elimination rules in Figure 3 ensure that terms depending on such a boundary can be “glued together” from terms depending on lower-dimensional simplices in the expected way. For instance, to define a term $a : A$ in context $\partial\Delta^1$ (i.e. in context $t : 2 \mid t \equiv 0 \vee t \equiv 1$), it is necessary and sufficient to give a term $a_0 : A$ in context $t : 2 \mid t \equiv 0$ and a term $a_1 : A$ in context $t : 2 \mid t \equiv 1$, such that if we assume $t \equiv 0 \wedge t \equiv 1$ then $a_0 \equiv a_1$. But the last requirement is vacuous, since $t \equiv 0 \wedge t \equiv 1 \vdash \perp$ so that in that context both reduce to rec_\perp . Moreover, since tope equality acts like judgmental equality, assuming $t : 2$ and $t \equiv 0$ is equivalent to assuming nothing at all, and similarly for assuming $t \equiv 1$.

Thus, a term $a : A$ in context $\partial\Delta^1$ is equivalently two terms $a_0, a_1 : A$ in no shape context, so that $\partial\Delta^1$ behaves like $\mathbf{2}$, the boolean type $\mathbf{1} + \mathbf{1}$. Similarly, a term $a : A$ in context $\partial\Delta^2$ is equivalently three terms $a_0, a_1, a_2 : A$ in context $t : 2$ such that $a_0[0/t] \equiv a_1[0/t]$ and $a_0[1/t] \equiv a_2[0/t]$ and $a_1[1/t] \equiv a_2[1/t]$, i.e. a “noncommutative triangle”.

More interestingly, $\Delta^1 \times \Delta^1$ (i.e. the shape $\{t : 2 \times 2 \mid \top\}$) behaves like the pushout of two copies of Δ^2 along their common diagonal boundary $\Delta^1_1 := \{\langle t, s \rangle : 2 \times 2 \mid t \equiv s\}$. For since we have $t : 2, s : 2 \vdash (t \leq s) \vee (s \leq t)$, a term $a : A$ in context $\Delta^1 \times \Delta^1$ is equivalently a term $a_0 : A$ in context $t : 2, s : 2 \mid (t \leq s)$ (which, up to tupling and permutation of variables, is just Δ^2) and a term $a_1 : A$ in context $t : 2, s : 2 \mid (s \leq t)$ (another copy of Δ^2), such that if we assume $t \leq s$ and $s \leq t$ then $a_0 \equiv a_1$. But $(t \leq s), (s \leq t) \vdash t \equiv s$, so this context is just a copy of Δ^1 , embedded into the two copies of Δ^2 as one of the boundary edges.

As an example application of this, recall that in Remark 3.2 we remarked that the cubical de Morgan algebra structure on shapes⁶ enables the construction of “connection” squares with the following faces, for any arrow f from x to y :

$$\begin{array}{ccc} \begin{array}{ccc} x & \xrightarrow{f} & y \\ f \downarrow & \searrow f & \cdot \\ y & \xrightarrow{\quad} & y \end{array} & \text{and} & \begin{array}{ccc} x & \xrightarrow{\quad} & x \\ \cdot & \searrow f & \downarrow f \\ x & \xrightarrow{f} & y \end{array} \end{array} \quad (3.4)$$

⁶Actually, a lattice structure suffices.

We denote these squares by \mathbb{V}_f and $\mathbb{\Lambda}_f$, respectively; in terms of the lattice operations $\mathbb{\Lambda}$ and \mathbb{V} they are defined by $\mathbb{V}_f(t, s) = f(t \mathbb{V} s)$ and $\mathbb{\Lambda}_f(t, s) = f(t \mathbb{\Lambda} s)$. Unlike in the cubical type theory of [5], we have not assumed \mathbb{V} and $\mathbb{\Lambda}$ as operations in our algebra of cubes, but we can nevertheless construct \mathbb{V}_f and $\mathbb{\Lambda}_f$ using $\text{rec}_{\mathbb{V}}$. Essentially, this is because a totally ordered set is automatically a lattice.

Proposition 3.5. *For any $f : 2 \rightarrow A$, we have squares $\mathbb{V}_f, \mathbb{\Lambda}_f : 2 \times 2 \rightarrow A$ with the faces displayed in (3.4), i.e. such that*

$$\begin{array}{ll} \mathbb{V}_f(0, s) \equiv f(s) & \mathbb{\Lambda}_f(0, s) \equiv f(0) \\ \mathbb{V}_f(t, 0) \equiv f(t) & \mathbb{\Lambda}_f(t, 0) \equiv f(0) \\ \mathbb{V}_f(1, s) \equiv f(1) & \mathbb{\Lambda}_f(0, s) \equiv f(s) \\ \mathbb{V}_f(t, 1) \equiv f(1) & \mathbb{\Lambda}_f(t, 0) \equiv f(t) \\ \mathbb{V}_f(t, t) \equiv f(t) & \mathbb{\Lambda}_f(t, t) \equiv f(t). \end{array}$$

Proof. We define

$$\begin{array}{l} \mathbb{V}_f(t, s) := \text{rec}_{\mathbb{V}}^{t \leq s, s \leq t}(f(s), f(t)) \\ \mathbb{\Lambda}_f(t, s) := \text{rec}_{\mathbb{V}}^{t \leq s, s \leq t}(f(t), f(s)). \end{array}$$

In both cases, if $t \leq s$ and $s \leq t$, then $t \equiv s$ and so $f(s) \equiv f(t)$, so the compatibility condition is satisfied. Geometrically, \mathbb{V}_f glues two copies of the degenerate 2-simplex $\lambda t. \lambda s. f(t)$ along their common 1-face, while $\mathbb{\Lambda}_f$ similarly glues two copies of the other degenerate 2-simplex $\lambda t. \lambda s. f(s)$. \square

As a second application, we observe that, at least as far as maps out of it are concerned, we may suppose Δ^2 to be a retract of $\Delta^1 \times \Delta^1$. This will be useful in a number of places to deduce 2-simplex information from 1-simplex assumptions.

Proposition 3.6. *For any type A , the type $\Delta^2 \rightarrow A$ is a retract of $\Delta^1 \times \Delta^1 \rightarrow A$.*

Proof. The retraction is easy:⁷

$$\lambda f. \lambda \langle t, s \rangle. f(t, s) : (\Delta^1 \times \Delta^1 \rightarrow A) \rightarrow (\Delta^2 \rightarrow A).$$

The section is where we have to use $\text{rec}_{\mathbb{V}}$:

$$\lambda f. \lambda \langle t, s \rangle. \text{rec}_{\mathbb{V}}^{t \leq s, s \leq t}(f(t, t), f(t, s)) : (\Delta^2 \rightarrow A) \rightarrow (\Delta^1 \times \Delta^1 \rightarrow A)$$

Again, if $t \leq s$ and $s \leq t$ then $t \equiv s$ so $f(t, t) \equiv f(t, s)$, so the compatibility condition holds. And if $\langle t, s \rangle : \Delta^2$ then $s \leq t$, so the composite of section followed by retraction is the identity. \square

Similar arguments apply in higher dimensions. For instance, the 3-dimensional ‘‘prism’’ $\Delta^2 \times \Delta^1 \equiv \{\langle \langle t_1, t_2 \rangle, t_3 \rangle \mid t_2 \leq t_1\}$ can be written as the union of three 3-simplices

$$\begin{array}{l} \Delta^3 = \{\langle \langle t_1, t_2 \rangle, t_3 \rangle \mid t_3 \leq t_2 \leq t_1\} \\ \Delta^3 = \{\langle \langle t_1, t_2 \rangle, t_3 \rangle \mid t_2 \leq t_3 \leq t_1\} \\ \Delta^3 = \{\langle \langle t_1, t_2 \rangle, t_3 \rangle \mid t_2 \leq t_1 \leq t_3\} \end{array}$$

⁷The apparently trivial η -expansion serves to create an element of an extension type with different domain.

along their common boundary 2-simplices

$$\begin{aligned}\Delta^2 &= \{\langle\langle t_1, t_2 \rangle, t_3 \rangle \mid t_3 \equiv t_2 \leq t_1\} \\ \Delta^2 &= \{\langle\langle t_1, t_2 \rangle, t_3 \rangle \mid t_2 \leq t_3 \equiv t_1\}.\end{aligned}$$

This enables us to show:

Proposition 3.7. *For any type A , the type $\Delta^3 \rightarrow A$ is a retract of $\Delta^2 \times \Delta^1 \rightarrow A$.*

Proof. There are actually many such retractions. To illustrate the above decomposition, we describe one that isn't the simplest. For the retraction we evaluate on the “middle” 3-simplex of the prism:

$$\lambda f. \lambda \langle t_1, t_2, t_3 \rangle. f \langle \langle t_1, t_3 \rangle, t_2 \rangle : (\Delta^2 \times \Delta^1 \rightarrow A) \rightarrow (\Delta^3 \rightarrow A).$$

This is well-defined since in Δ^3 we have $t_3 \leq t_2 \leq t_1$, hence in particular $t_3 \leq t_1$. The section is defined using a triple rec_\vee , which we can write informally as a case split:

$$\lambda f. \lambda \langle \langle t_1, t_2 \rangle, t_3 \rangle. \begin{cases} f(t_1, t_2, t_2) & t_3 \leq t_2 \\ f(t_1, t_3, t_2) & t_2 \leq t_3 \leq t_1 \\ f(t_1, t_1, t_2) & t_1 \leq t_3 \end{cases}$$

Here in all cases we have $t_2 \leq t_1$, so in each case the requirement is met for f to be defined. The agreement on the boundary 2-simplices, when $t_3 \equiv t_1$ or $t_3 \equiv t_2$, is also obvious, as is the fact that this is a section of the above retraction. \square

3.3 Joins of simplices In this paper we will only need shapes and simplices of very small dimension such as $n = 2, 3$. However, in this subsection we briefly indicate how some important shapes such as n -dimensional simplicial spheres and horns can be defined, using an analogue of Joyal’s non-symmetric monoidal “join” operation [9].

Informally, if given a pair of shapes

$$\{t : \mathcal{2}^n \mid \phi\} \quad \text{and} \quad \{s : \mathcal{2}^m \mid \psi\},$$

in the n -cube and m -cube respectively, their *join* is the shape in the $(n + 1 + m)$ -cube whose tope is those $\langle t_1, \dots, t_n, u, s_1, \dots, s_m \rangle : \mathcal{2}^n \times \mathcal{2} \times \mathcal{2}^m$ satisfying $\phi[u/0] \wedge \psi[u/1]$, i.e. satisfying ϕ except with u substituted for all occurrences of the term 0 and also ψ with u substituted for all occurrences of the term 1. However, this sort of “substitution of a variable for a constant” is not technically possible, so instead we construct joins as restrictions of an auxiliary “gluing” operation on shapes in cubes of a larger dimension.

Definition 3.8. Given shapes

$$A := \{t : \mathcal{2}^{1+n+1} \mid \phi\} \quad \text{and} \quad B := \{s : \mathcal{2}^{1+m+1} \mid \psi\} \tag{3.9}$$

where we write $t \equiv \langle t_-, t_1, \dots, t_n, t_+ \rangle : \mathcal{2}^{1+n+1}$ and $s \equiv \langle s_-, s_1, \dots, s_m, s_+ \rangle : \mathcal{2}^{1+m+1}$, their **gluing** is the shape $A \otimes B$ in $\mathcal{2}^{1+n+1+m+1}$ defined by

$$A \otimes B := \{\langle t_-, t_1, \dots, t_n, u, s_1, \dots, s_m, s_+ \rangle : \mathcal{2}^{1+n+1+m+1} \mid \phi[u/t_+] \wedge \psi[u/s_-]\}$$

Example 3.10. For any $n, m \geq 0$, the gluing $\Delta^{1+n+1} \otimes \Delta^{1+m+1}$ of simplices is the simplex $\Delta^{1+n+1+m+1}$.

Definition 3.11. Given a shape

$$A := \{\langle t_-, t_1, \dots, t_n, t_+ \rangle : 2^{1+n+1} \mid \phi\},$$

its **restriction** is the shape

$$\bar{A} := \{\langle t_1, \dots, t_n \rangle : 2^n \mid \phi[1/t_-, 0/t_+]\}.$$

Example 3.12. For any $n \geq 0$, the n -simplex is the restriction of the $n + 2$ -simplex.

Definition 3.13 (join). An **augmented shape** is a shape in 2^n of the form \bar{A} for some specified shape A in 2^{1+n+1} . In other words, an augmented shape is really just an arbitrary shape in 2^{1+n+1} , but we regard it as its restriction in 2^n equipped with extra data.

Now given augmented shapes \bar{A} in 2^n and \bar{B} in 2^m , their **join** is the augmented shape defined as the restriction of the gluing:

$$\bar{A} \star \bar{B} := \overline{A \otimes B}.$$

Example 3.14 (joins of simplices). The simplex Δ^n has a canonical augmentation as $\Delta^n = \overline{\Delta^{1+n+1}}$; we always regard it as augmented in this way. Thus the join $\Delta^n \star \Delta^m$ is defined, and since $\Delta^{1+n+1} \otimes \Delta^{1+m+1} = \Delta^{1+n+1+m+1}$ we have $\Delta^n \star \Delta^m = \Delta^{n+1+m}$. In particular, we can “construct” the simplices by iterated joins as $\Delta^n := \Delta^{n-1} \star \Delta^0$, although this is not really a definition since to augment Δ^{n-1} we have to already have Δ^{n+1} .

Definition 3.15 (pushout join). Given inclusions of augmented shapes $A \subseteq B$ and $C \subseteq D$, their **pushout join** is

$$A \hat{\star} B = (A \star D) \cup (B \star C) \subseteq B \star D$$

Example 3.16 (boundaries of simplices). We can define the simplex boundaries (simplicial spheres) $\partial\Delta^n$ by recursive pushout joins. As base cases we define the augmented 0- and 1-spheres:

$$\begin{aligned} \partial\Delta^0 &= \overline{\{\langle t_-, t_+ \rangle : 2^2 \mid t_+ \equiv t_-\}} \subseteq \Delta^0 \\ \partial\Delta^1 &= \overline{\{\langle t_-, t_1, t_+ \rangle : 2^3 \mid (t_+ \equiv t_1 \leq t_-) \vee (t_+ \leq t_1 \equiv t_-)\}} \subseteq \Delta^1 \end{aligned}$$

and then define recursively

$$\partial\Delta^{n+1} = \partial\Delta^n \hat{\star} \partial\Delta^0 \subseteq \Delta^{n+1}.$$

For instance, we have

$$\begin{aligned} \partial\Delta^1 \star \Delta^0 &= \overline{\{\langle t_-, t_1, u, s_+ \rangle : 2^4 \mid (s_+ \leq u \equiv t_1 \leq t_-) \vee (s_+ \leq u \leq t_1 \equiv t_-)\}} \\ \Delta^1 \star \partial\Delta^0 &= \overline{\{\langle t_-, t_1, u, s_+ \rangle : 2^4 \mid s_+ \equiv u \leq t_1 \leq t_-\}} \end{aligned}$$

and hence

$$\begin{aligned} \partial\Delta^2 &= (\partial\Delta^1 \star \Delta^0) \cup (\Delta^1 \star \partial\Delta^0) = \\ &= \overline{\{\langle t_-, t_1, u, s_+ \rangle : 2^4 \mid (s_+ \leq u \equiv t_1 \leq t_-) \vee (s_+ \leq u \leq t_1 \equiv t_-) \vee (s_+ \equiv u \leq t_1 \leq t_-)\}} \end{aligned}$$

Restricting by substituting $s_+ \equiv 0$ and $t_- \equiv 1$, we obtain the usual definition of the boundary of the 2-simplex:

$$\{\langle t_1, u \rangle : 2^2 \mid (0 \leq u \equiv t_1 \leq 1) \vee (0 \leq u \leq t_1 \equiv 1) \vee (0 \equiv u \leq t_1 \leq 1)\}$$

4. Equivalences involving extension types

In this section we collect several important equivalences involving extension types, mainly straightforward generalizations of standard facts about dependent function types. Moreover, since our extension types, Π -types, and Σ -types have judgmental η -conversion as well as β -reduction, all of these equivalences are actually “judgmental isomorphisms”, i.e. the composites in both directions are judgmentally equal to the identity. We also formulate a function extensionality axiom for extension types.

Note that when a theorem is stated as an equivalence between two types $A \simeq B$, we will not scruple to later use the *specific* equivalence constructed in its proof rather than the mere existence of such an equivalence. This is in accord with the “proof-relevant” philosophy of type theory and the propositions-as-types principle: a proof of a theorem is the construction of an inhabitant of some type, in this case a type of equivalences.

4.1 Commutation of arguments and currying For ordinary function types we have an equivalence $(X \rightarrow (Y \rightarrow Z)) \simeq (Y \rightarrow (X \rightarrow Z))$, and similarly in the dependent case we have $(\prod_{x:X} \prod_{y:Y} Z(x, y)) \simeq (\prod_{y:Y} \prod_{x:X} Z(x, y))$. The following theorems are analogues of this for extension types.

Theorem 4.1. *If $t : I \mid \phi \vdash \psi$ and $X : \mathcal{U}$, while $Y : \{t : I \mid \psi\} \rightarrow X \rightarrow \mathcal{U}$ and $f : \prod_{t:I|\phi} \prod_{x:X} Y(t, x)$, then*

$$\left\langle \prod_{t:I|\phi} \left(\prod_{x:X} Y(t, x) \right) \Big|_f^\phi \right\rangle \simeq \prod_{x:X} \left\langle \prod_{t:I|\phi} Y(t, x) \Big|_{\lambda t. f(t, x)}^\phi \right\rangle.$$

Note that X cannot depend on t or ψ , since otherwise the right-hand side of the equivalence would be ill-formed.

Proof. As for ordinary dependent functions, this is just application and re-abstraction: from left to right $g \mapsto \lambda x. \lambda t. g(t, x)$, and from right to left $h \mapsto \lambda t. \lambda x. h(x, t)$. We just have to verify that the requisite judgmental equations hold: if $g(t) \equiv f(t)$ assuming ϕ , then $g(t, x) \equiv f(t, x)$ assuming ϕ ; while if $h(x, t) \equiv f(t, x)$ assuming ϕ , then $\lambda t. \lambda x. h(x, t) \equiv \lambda t. \lambda x. f(t, x) \equiv f$ assuming ϕ (using η -conversion). The composites in both directions are judgmentally the identity, by η -conversion. \square

Theorem 4.2. *If $t : I \mid \phi \vdash \psi$ and $s : J \mid \chi \vdash \zeta$, while*

$$X : \{t : I \mid \psi\} \rightarrow \{s : J \mid \zeta\} \rightarrow \mathcal{U}$$

and $f : \prod_{(t,s):I \times J | (\phi \wedge \zeta) \vee (\psi \wedge \chi)} X(t, s)$, then

$$\begin{aligned} & \left\langle \prod_{t:I|\psi} \left\langle \prod_{s:J|\zeta} X(t, s) \Big|_{\lambda s. f(t, s)}^\chi \right\rangle \Big|_{\lambda t. \lambda s. f(t, s)}^\phi \right\rangle \\ & \simeq \left\langle \prod_{(t,s):I \times J | \psi \wedge \zeta} X(t, s) \Big|_f^{(\phi \wedge \zeta) \vee (\psi \wedge \chi)} \right\rangle \\ & \simeq \left\langle \prod_{s:J|\zeta} \left\langle \prod_{t:I|\psi} X(t, s) \Big|_{\lambda t. f(t, s)}^\phi \right\rangle \Big|_{\lambda s. \lambda t. f(t, s)}^\chi \right\rangle. \end{aligned}$$

The equivalence of the sides to the middle in Theorem 4.2 is a version of *currying*. The shape $\{(t, s) : I \times J \mid (\phi \wedge \zeta) \vee (\psi \wedge \chi)\}$ may be called the **pushout product** of the two inclusions $\{t : I \mid \phi\} \subseteq \{t : I \mid \psi\}$ and $\{s : J \mid \chi\} \subseteq \{s : J \mid \zeta\}$.

Proof. We first check the well-formedness of the extension types. Whenever $t : I$ is such that ψ holds, we have for each $s : J$ such that χ holds a term $f\langle t, s \rangle : X(t, s)$, defining a function $\lambda s. f\langle t, s \rangle : \prod_{s:J|\chi} X(t, s)$; thus we can form $\left\langle \prod_{s:J|\chi} X \Big|_{\lambda s. f\langle t, s \rangle}^{\chi} \right\rangle$. Now whenever $t : I$ is such that ϕ holds, we have for each $s : J$ such that ζ holds a term $f\langle t, s \rangle : X(t, s)$, which of course equals the first $f\langle t, s \rangle$ if we also have χ , so we have the function $\lambda t. \lambda s. f\langle t, s \rangle : \prod_{t:I|\phi} \left\langle \prod_{s:J|\zeta} X \Big|_{\lambda s. f\langle t, s \rangle}^{\chi} \right\rangle$ and thus can form the left-hand side. The right-hand side is dual, and the middle is easy. Now the equivalence between the left- and right-hand types is again just application and re-abstraction, while the equivalence of both to the middle type is ordinary currying. \square

Recall that when ϕ or χ is \perp , extension types behave like ordinary (dependent) function types, so we omit the angle brackets from the notation as in (2.3), (2.4). Thus, as special cases of Theorem 4.2 we have

$$\begin{aligned} \left\langle \prod_{t:I|\psi} \left(\prod_{s:J|\zeta} X(t, s) \right) \Big|_{\lambda t. \lambda s. f\langle t, s \rangle}^{\phi} \right\rangle &\simeq \left\langle \prod_{(t,s):I \times J|\psi \wedge \zeta} X(t, s) \Big|_f^{\phi \wedge \zeta} \right\rangle \\ &\simeq \prod_{s:J|\zeta} \left\langle \prod_{t:I|\psi} X(t, s) \Big|_{\lambda s. f\langle t, s \rangle}^{\phi} \right\rangle. \end{aligned}$$

and

$$\begin{aligned} \prod_{t:I|\psi} \prod_{s:J|\zeta} X(t, s) &\simeq \prod_{(t,s):I \times J|\psi \wedge \zeta} X(t, s) \\ &\simeq \prod_{s:J|\zeta} \prod_{t:I|\psi} X(t, s). \end{aligned}$$

with further notational specializations to the non-dependent case, such as

$$\begin{aligned} (\{t : I \mid \psi\} \rightarrow (\{s : J \mid \zeta\} \rightarrow X)) &\simeq (\{t : I \mid \psi\} \times \{s : J \mid \zeta\} \rightarrow X) \\ &\simeq (\{s : J \mid \zeta\} \rightarrow (\{t : I \mid \psi\} \rightarrow X)). \end{aligned}$$

4.2 Extending into Σ -types (the non-axiom of choice) For ordinary dependent functions we have the following equivalence [29, Theorem 2.15.7]:

$$\left(\prod_{x:X} \sum_{y:Y(x)} Z(x, y) \right) \simeq \left(\sum_{f:\prod_{x:X} Y(x)} \prod_{x:X} Z(x, f(x)) \right)$$

The following is a version of this for extension types.

Theorem 4.3. *If $t : I \mid \phi \vdash \psi$, while $X : \{t : I \mid \psi\} \rightarrow \mathcal{U}$ and $Y : \prod_{t:I|\psi} (X \rightarrow \mathcal{U})$, while $a : \prod_{t:I|\phi} X(t)$ and $b : \prod_{t:I|\phi} Y(t, x(t))$, then*

$$\left\langle \prod_{t:I|\psi} \left(\sum_{x:X(t)} Y(t, x) \right) \Big|_{\lambda t. (a(t), b(t))}^{\phi} \right\rangle \simeq \sum_{f:\left\langle \prod_{t:I|\psi} X(t) \Big|_a^{\phi} \right\rangle} \left\langle \prod_{t:I|\psi} Y(t, f(t)) \Big|_b^{\phi} \right\rangle.$$

Proof. As in the ordinary case, this is just composing the introduction and elimination rules. From left to right, $h \mapsto (\lambda t. \pi_1(h(t)), \lambda t. \pi_2(h(t)))$; while from right to left, $(f, g) \mapsto \lambda t. (f(t), g(t))$. It is easy to check that the required judgmental equalities are preserved, and the β - and η -conversion rules make these inverse equivalences. \square

4.3 Composites and unions of cofibrations These equivalences have no analogue for ordinary dependent functions.

Theorem 4.4. *Suppose $t : I \mid \phi \vdash \psi$ and $t : I \mid \psi \vdash \chi$, and that $X : \{t : I \mid \chi\} \rightarrow \mathcal{U}$ and $a : \prod_{t:I|\phi} X(t)$. Then*

$$\left\langle \prod_{t:I|\chi} X \Big|_a^\phi \right\rangle \simeq \left(\sum_{f : \left\langle \prod_{t:I|\psi} X \Big|_a^\phi \right\rangle} \left\langle \prod_{t:I|\chi} X \Big|_f^\psi \right\rangle \right)$$

Proof. From left to right, $h \mapsto (\lambda t. h(t), \lambda t. h(t))$; the η -expansions indicate a re-packaging of the same term into a different extension type. Similarly, from right to left, $(f, g) \mapsto \lambda t. g(t)$. \square

Theorem 4.5. *Suppose $t : I \vdash \phi$ **tope** and $t : I \vdash \psi$ **tope**, and that we have $X : \{t : I \mid \phi \vee \psi\} \rightarrow \mathcal{U}$ and $a : \prod_{t:I|\psi} X(t)$. Then*

$$\left\langle \prod_{t:I|\phi \vee \psi} X \Big|_a^\psi \right\rangle \simeq \left\langle \prod_{t:I|\phi} X \Big|_{\lambda t. a(t)}^{\phi \wedge \psi} \right\rangle.$$

Proof. From left to right this is just re-packaging, $h \mapsto \lambda t. h(t)$. From right to left is a little less obvious: $g \mapsto \lambda t. \text{rec}_{\vee}^{\phi, \psi}(g(t), a(t))$, which is well-defined since $g(t) \equiv a(t)$ for $t : I$ satisfying $\phi \wedge \psi$. The required equalities are immediate, and the composites are the identity by the β - and η -conversion rules for rec_{\vee} . \square

4.4 Relative function extensionality We will need to assume a function extensionality axiom for extension types, with respect to the homotopical identity types, which we write as equalities $x = y$. In homotopy type theory there are at least three formulations of function extensionality for ordinary dependent functions, which turn out to be equivalent:

- For $f, g : \prod_{x:A} B(x)$, if $\prod_{x:A} (fx = gx)$, then $f = g$.
- For $f, g : \prod_{x:A} B(x)$, the canonical map $(f = g) \rightarrow \prod_{x:A} (fx = gx)$ is an equivalence.
- If each $B(x)$ is contractible, then so is $\prod_{x:A} B(x)$.

The first is a naïve statement of function extensionality uninformed by homotopy theory; the second is a homotopical refinement of it; and the third is an easy consequence of the second that was observed by Voevodsky to be equivalent to it. The equivalence in the second statement is meant in the usual sense of [29, §4.5].

We do not know whether the analogues of these three formulations for extension types are still equivalent, so as our axiom of function extensionality we take one that we do know how to prove the others from. Somewhat surprisingly, this is the *third* rather than the second.

Axiom 4.6 (relative function extensionality⁸). *Supposing $t : I \mid \phi \vdash \psi$ and that $A : \{t : I \mid \psi\} \rightarrow \mathcal{U}$ is such that each $A(t)$ is contractible, and moreover $a : \prod_{t:I|\phi} A(t)$, then $\left\langle \prod_{t:I|\psi} A(t) \Big|_a^\phi \right\rangle$ is contractible.*

Now suppose given any $A : \{t : I \mid \psi\} \rightarrow \mathcal{U}$ and $a : \prod_{t:I|\phi} A(t)$, and also $f, g : \left\langle \prod_{t:I|\psi} A \Big|_a^\phi \right\rangle$. Then in the context of $t : I$ and ψ we can form the identity type $f(t) = g(t)$, and thereby the extension type $\left\langle \prod_{t:I|\psi} f(t) = g(t) \Big|_{\lambda t. \text{refl}}^\phi \right\rangle$. Of course, we have $\lambda t^{I|\psi}. \text{refl} : \left\langle \prod_{t:I|\psi} f(t) = f(t) \Big|_{\lambda t. \text{refl}}^\phi \right\rangle$, so by identity elimination, we obtain a map

$$(f = g) \rightarrow \left\langle \prod_{t:I|\psi} f(t) = g(t) \Big|_{\lambda t. \text{refl}}^\phi \right\rangle. \quad (4.7)$$

Analogously to ordinary function extensionality, we have:

Proposition 4.8. *Assuming Axiom 4.6:*

⁸Or “extension extensionality”.

(i) The map (4.7) is an equivalence.

(ii) In particular, for any $f, g : \langle \prod_{t:I|\psi} A \Big|_a^\phi \rangle$, if $\langle \prod_{t:I|\psi} f(t) = g(t) \Big|_{\lambda t. \text{refl}}^\phi \rangle$ then $f = g$.

Proof. It suffices to prove that for each f the induced map on total spaces

$$\left(\sum_{g: \langle \prod_{t:I|\psi} A(t) \Big|_a^\phi \rangle} (f = g) \right) \rightarrow \left(\sum_{g: \langle \prod_{t:I|\psi} A(t) \Big|_a^\phi \rangle} \langle \prod_{t:I|\psi} f(t) = g(t) \Big|_{\lambda t. \text{refl}}^\phi \rangle \right)$$

is an equivalence. But the domain of this map is contractible, as a based path space, so it suffices to prove that its codomain is also contractible. However, by Theorem 4.3 this codomain is equivalent to

$$\left\langle \prod_{t:I|\psi} \left(\sum_{y:A(t)} (f(t) = y) \right) \Big|_{\lambda t. (a, \text{refl})}^\phi \right\rangle$$

which is contractible by Axiom 4.6, since each $\sum_{y:A(t)} (f(t) = y)$ is a based path space and hence contractible. \square

Remark 4.9. Note that the identity type $f = g$ appearing as the domain of (4.7) refers to identity *in* the extension type $\langle \prod_{t:I|\psi} A(t) \Big|_{\lambda t. a}^\phi \rangle$, and Proposition 4.8(i) identifies this with a type of “relative homotopies” or “relative pointwise equalities” that must restrict to reflexivity on the domain of the cofibration. This explains the name “relative function extensionality”. Note also that when ϕ is \perp , relative function extensionality reduces to ordinary function extensionality for (dependent) function types whose domain is a shape rather than a type.

Another important consequence of Axiom 4.6 is the following.

Proposition 4.10 (homotopy extension property). *Let $t : I \mid \phi \vdash \psi$. Assuming Axiom 4.6, if we have $A : \{t : I \mid \psi\} \rightarrow \mathcal{U}$ and $b : \prod_{t:I|\psi} A(t)$, and moreover $a : \prod_{t:I|\phi} A(t)$ and $e : \prod_{t:I|\phi} a(t) = b(t)$, then we have $a' : \langle \prod_{t:I|\psi} A(t) \Big|_a^\phi \rangle$ and $e' : \langle \prod_{t:I|\psi} a'(t) = b(t) \Big|_e^\phi \rangle$.*

Proof. The extension type $\langle \prod_{t:I|\psi} \left(\sum_{y:A(t)} (y = b(t)) \right) \Big|_{\lambda t. (a(t), e(t))}^\phi \rangle$ is contractible by Axiom 4.6, hence inhabited. We obtain a' and e' by applying Theorem 4.3. \square

We do not know how to derive Axiom 4.6 or Proposition 4.10 assuming only the conclusions of Proposition 4.8(i) or (ii), but we can show that Proposition 4.8(ii) and Proposition 4.10 imply Axiom 4.6.

Proposition 4.11. *If Proposition 4.8(ii) and the homotopy extension property hold, then the relative function extensionality axiom holds.*

Proof. Suppose $A : \{t : I \mid \psi\} \rightarrow \mathcal{U}$ and $a : \prod_{t:I|\phi} A(t)$ such that each $A(t)$ is contractible. The latter assumption supplies centers of contraction $b(t)$ for each $t : I$ such that ψ , hence a function $b : \prod_{t:I|\psi} A(t)$. Contractibility of each $A(t)$ also shows that if ϕ then $a(t) = b(t)$, hence $e : \prod_{t:I|\phi} a(t) = b(t)$. Thus, by Proposition 4.10, we have $a' : \langle \prod_{t:I|\psi} A(t) \Big|_a^\phi \rangle$ and $e' : \langle \prod_{t:I|\psi} a'(t) = b(t) \Big|_e^\phi \rangle$.

It remains to show that any $f : \langle \prod_{t:I|\psi} A(t) \Big|_a^\phi \rangle$ is equal to a' . By Proposition 4.8(ii), for this it suffices to inhabit $\langle \prod_{t:I|\psi} f(t) = a'(t) \Big|_{\lambda t. \text{refl}}^\phi \rangle$. Now since each $A(t)$ is contractible, we have $c : \prod_{t:I|\psi} f(t) = a'(t)$, and moreover if ϕ then $c(t) = \text{refl}$ since any two paths in a contractible type are equal. Thus, applying Proposition 4.10 to $\lambda t. f(t) = a'(t)$ in place of A , with c in place of b and $\lambda t. \text{refl}$ in place of a , we have an element of $\langle \prod_{t:I|\psi} f(t) = a'(t) \Big|_{\lambda t. \text{refl}}^\phi \rangle$ as desired. \square

A similar argument and an induction on n shows:

Proposition 4.12. *Assuming Axiom 4.6, if $A : \{t : I \mid \psi\} \rightarrow \mathcal{U}$ and $a : \prod_{t:I|\phi} A(t)$ are such that each $A(t)$ is an n -type, then $\left\langle \prod_{t:I|\psi} A(t) \Big|_a^\phi \right\rangle$ is also an n -type. \square*

For the rest of the paper, we will assume relative function extensionality, Axiom 4.6, without further comment.

5. Segal types

The simplices defined in §3 are used to parametrize internal categorical structure in types satisfying an analogue of the famous Segal condition. Interestingly, because we express the Segal condition in the internal language, it has a more compact form than usual. We first introduce notation for “hom” types of various dimensions whose terms are “morphisms” or “compositions” in another type. We then state our Segal type axiom and prove the somewhat surprising fact that a single low-dimensional condition suffices to establish the expected categorical properties.

5.1 The Segal condition We introduce the following notation.

Definition 5.1. Given $x, y : A$, determining a term $[x, y] : A$ in context $\partial\Delta^1$, we define

$$\text{hom}_A(x, y) := \left\langle \Delta^1 \rightarrow A \Big|_{[x,y]}^{\partial\Delta^1} \right\rangle.$$

We refer to an element of $\text{hom}_A(x, y)$ as an **arrow** from x to y in A .

This plays the role of the directed hom-space of A . Note that every $f : \text{hom}_A(x, y)$ is a kind of *function* from 2 to A , with the property that $f(0) \equiv x$ and $f(1) \equiv y$.

Definition 5.2. Similarly, for $x, y, z : A$ and $f : \text{hom}_A(x, y)$, $g : \text{hom}_A(y, z)$, and $h : \text{hom}_A(x, z)$ we have an induced term $[x, y, z, f, g, h] : A$ in context $\partial\Delta^2$, and thereby an extension type that we denote

$$\text{hom}_A^2 \left(\begin{array}{ccc} & f & y & g & \\ & \nearrow & & \searrow & \\ x & & & & z \\ & \nwarrow & h & \nearrow & \end{array} \right) := \left\langle \Delta^2 \rightarrow A \Big|_{[x,y,z,f,g,h]}^{\partial\Delta^2} \right\rangle.$$

or $\text{hom}_A^2(f, g; h)$ when space is at a premium.

In a few places we will use 3-simplices with specified boundaries, but we will not introduce a particular notation for them. We will not have any need for n -simplices with $n > 3$.

Definition 5.3. A **Segal type** is a type A such that for all $x, y, z : A$ and $f : \text{hom}_A(x, y)$ and $g : \text{hom}_A(y, z)$, the type

$$\sum_{h:\text{hom}_A(x,z)} \text{hom}_A^2 \left(\begin{array}{ccc} & f & y & g & \\ & \nearrow & & \searrow & \\ x & & & & z \\ & \nwarrow & h & \nearrow & \end{array} \right) \tag{5.4}$$

is contractible.

In particular, (5.4) is inhabited, and the first component of this inhabitant we call $g \circ f : \text{hom}_A(x, z)$, the **composite** of g and f . The second component of this inhabitant is a 2-simplex in $\text{hom}_A^2(f, g; g \circ f)$, which we consider a “witness that $g \circ f$ is the composite of g and f ”; we denote it by $\text{comp}_{g,f}$. The contractibility of (5.4) implies that composites are unique in the following

sense: given $h : \text{hom}_A(x, z)$ and any witness $p : \text{hom}_A^2(f, g; h)$, we have $(h, p) = (g \circ f, \text{comp}_{g,f})$, and hence in particular $h = g \circ f$.

We can usefully reformulate (5.4) as a single extension type of functions $\Delta^2 \rightarrow A$ that restrict to f and g on the 2-1-horn defined by

$$\Lambda_1^2 = \{\langle s, t \rangle : 2 \times 2 \mid s = 1 \vee t = 0\}.$$

This observation is the key step in the proof of the following alternate characterization of Segal types (cf. Proposition A.21).

Theorem 5.5. *A type A is Segal if and only if the restriction map*

$$(\Delta^2 \rightarrow A) \rightarrow (\Lambda_1^2 \rightarrow A)$$

is an equivalence.

Proof. If Δ_1^1 denotes the diagonal 1-face $\{\langle s, t \rangle : 2 \times 2 \mid s = t\}$ of Δ^2 , then we have $\Lambda_1^2 \cap \Delta_1^1 = \partial \Delta_1^1$ and $\Lambda_1^2 \cup \Delta_1^1 = \partial \Delta^2$. Therefore, by Theorem 4.5, to extend a map $\Lambda_1^2 \rightarrow A$ to $\partial \Delta^2$ is equivalent to extending its restriction to $\partial \Delta_1^1$ to Δ_1^1 . This provides the second in the following chain of equivalences; the third is Theorem 4.4.

$$\begin{aligned} \sum_{h: \text{hom}_A(x, z)} \text{hom}_A^2 \left(\begin{array}{ccc} & f & \\ x & \xrightarrow{y} & z \\ & h & \end{array} \right) &\equiv \sum_{h: \langle \Delta^1 \rightarrow A \big|_{[x, z]}^{\partial \Delta^1} \rangle} \langle \Delta^2 \rightarrow A \big|_{[x, y, z, f, g, h]}^{\partial \Delta^2} \rangle \\ &\simeq \sum_{\ell: \langle \partial \Delta^2 \rightarrow A \big|_{[x, y, z, f, g]}^{\Lambda_1^2} \rangle} \langle \Delta^2 \rightarrow A \big|_{\ell}^{\partial \Delta^2} \rangle \\ &\simeq \langle \Delta^2 \rightarrow A \big|_{[x, y, z, f, g]}^{\Lambda_1^2} \rangle. \end{aligned}$$

In other words, (5.4) is the type of functions $\Delta^2 \rightarrow A$ that restrict to f and g on the 2-1-horn. Definition 5.3 asserts that A is a Segal type if for any composable f and g there is a unique such extension.

Using Theorem 4.4 again, we have

$$(\Delta^2 \rightarrow A) \simeq \sum_{k: \Lambda_1^2 \rightarrow A} \langle \Delta^2 \rightarrow A \big|_k^{\Lambda_1^2} \rangle.$$

Therefore, $\Delta^2 \rightarrow A$ is the total space of a type family over $\Lambda_1^2 \rightarrow A$ whose fibers are the types (5.4). Since the projection from a total space is an equivalence exactly when all the fibers are contractible, the result follows. \square

Corollary 5.6. *If X is either a type or a shape and $A : X \rightarrow \mathcal{U}$ is such that each $A(x)$ is a Segal type, then the dependent function type $\prod_{x: X} A(x)$ is a Segal type.*

Proof. Applying Theorem 4.1 or Theorem 4.2 to rearrange function types, we have $(\Delta^2 \rightarrow \prod_{x: X} A(x)) \simeq \prod_{x: X} (\Delta^2 \rightarrow A(x))$ and similarly for Λ_1^2 . Since $\prod_{x: X}$ preserves fiberwise equivalences (using function extensionality or relative function extensionality), the result follows from Theorem 5.5. \square

In the rest of this section we show that Segal types behave like categories, or more precisely $(\infty, 1)$ -categories.

5.2 Identity Identities in a Segal type are obtained as constant maps.

Definition 5.7. For any $x : A$, define a term $\text{id}_x : \text{hom}_A(x, x)$ by $\text{id}_x(s) \equiv x$ for all $s : 2$.

The pair of “degenerate” 2-simplices witness that identities behave as identities in a Segal type:

Proposition 5.8. *If A is a Segal type with terms $x, y : A$, then for any $f : \text{hom}_A(x, y)$ we have $\text{id}_y \circ f = f$ and $f \circ \text{id}_x = f$.*

Proof. For any $f : \text{hom}_A(x, y)$ we have a canonical 2-simplex:

$$\lambda_{s,t}. f(s) : \text{hom}_A^2 \left(\begin{array}{ccc} & f & \\ x & \xrightarrow{f} & y \\ & f & \\ & & \text{id}_y \end{array} \right).$$

To check that this has the right boundary, we see that $(s, 0) \mapsto f(s)$ and $(s, s) \mapsto f(s)$, while $(1, t) \mapsto f(1) = y$. Thus, by uniqueness of composites, $\text{id}_y \circ f = f$; and similarly $f \circ \text{id}_x = f$. \square

5.3 Associativity We now prove that composition in a Segal type is associative. At first this may be surprising, since the definition of Segal type refers only to 2-simplices; but its “uniformity” allows us to apply it pointwise to arrows and use the fact that products of simplices contain higher-dimensional simplices.

Proposition 5.9. *If A is a Segal type with terms $x, y, z, w : A$, then for any $f : \text{hom}_A(x, y)$, $g : \text{hom}_A(y, z)$, $h : \text{hom}_A(z, w)$ we have $(h \circ g) \circ f = h \circ (g \circ f)$.*

Proof. By Corollary 5.6, if A is Segal then so is $A^2 := (2 \rightarrow A)$.⁹ Thus, for any $f : \text{hom}_A(x, y)$, $g : \text{hom}_A(y, z)$, $h : \text{hom}_A(z, w)$, the type

$$\sum_{p:\text{hom}_{A^2}(f,h)} \text{hom}_{A^2}^2 \left(\begin{array}{ccc} & \text{comp}_{g,f} & \\ f & \xrightarrow{g} & h \\ & p & \\ & & \text{comp}_{h,g} \end{array} \right)$$

is contractible, and in particular inhabited. Here $\text{comp}_{g,f}$ is abusive notation for the function $2 \rightarrow A^2$ that is built from two copies of $\text{comp}_{g,f} : \Delta^2 \rightarrow A$ using the equivalence $\Delta^1 \times \Delta^1 \simeq \Delta^2 \cup_{\Delta^1} \Delta^2$ discussed in §3.2.

The second component of this inhabitant is a 2-simplex witness $\Delta^2 \times 2 \rightarrow A$. There is a function

$$\lambda(t_1, t_2, t_3).((t_1, t_3), t_2) : \Delta^3 \rightarrow \Delta^2 \times 2$$

that picks out the “middle shuffle”. The 1st and 2nd faces are identified with further restrictions

$$\lambda(s, t).(s, s, t) : \Delta^2 \rightarrow \Delta^3 \quad \lambda(s, t).(s, t, t) : \Delta^2 \rightarrow \Delta^3,$$

with a common edge

$$\lambda t.(t, t, t) : \Delta^1 \rightarrow \Delta^3.$$

This edge defines an inhabitant $\ell : \text{hom}_A(x, w)$, while the pair of 2-simplices define witnesses that ℓ is the composite of $h \circ g$ and f , and that ℓ is the composite of h with $f \circ g$, respectively. In particular, $(h \circ g) \circ f = h \circ (g \circ f)$. \square

⁹For notational conciseness, we sometimes abbreviate the function type $X \rightarrow A$ as A^X , particularly in the case $X = 2$.

5.4 Homotopies Let A be a Segal type with terms $x, y : A$. Given two arrows $f, g : \text{hom}_A(x, y)$, there are two ways to say that f and g are the same:

- we might have a path $p : f =_{\text{hom}_A(x, y)} g$, or
- we might have a 2-simplex $q : \text{hom}_A^2 \left(\begin{array}{ccc} \text{id}_x & x & f \\ x & \xrightarrow{g} & y \end{array} \right)$.¹⁰

We demonstrate that these two types are in fact equivalent:

Proposition 5.10. *For any $f, g : \text{hom}_A(x, y)$ in a Segal type A , the natural map*

$$(f = g) \rightarrow \text{hom}_A^2 \left(\begin{array}{ccc} \text{id}_x & x & f \\ x & \xrightarrow{g} & y \end{array} \right) \quad (5.11)$$

is an equivalence.

Proof. The map (5.11) is defined by path induction, since for any f we have a “degenerate” 2-simplex

$$\lambda s, t. f(t) : \text{hom}_A^2 \left(\begin{array}{ccc} \text{id}_x & x & f \\ x & \xrightarrow{f} & y \end{array} \right)$$

defined to be constant on one input. To show that (5.11) is an equivalence, it suffices to show that the map of total spaces

$$\left(\sum_{g : \text{hom}_A(x, y)} (f = g) \right) \rightarrow \left(\sum_{g : \text{hom}_A(x, y)} \text{hom}_A^2 \left(\begin{array}{ccc} \text{id}_x & x & f \\ x & \xrightarrow{g} & y \end{array} \right) \right)$$

is an equivalence. But here both types are contractible; the first since it is a based path space, and the second since A is a Segal type. \square

More generally, we can say:

Proposition 5.12. *For $f : \text{hom}_A(x, y)$ and $g : \text{hom}_A(y, z)$ and $h : \text{hom}_A(x, z)$ in a Segal type A , the natural map*

$$(g \circ f = h) \rightarrow \text{hom}_A^2 \left(\begin{array}{ccc} f & y & g \\ x & \xrightarrow{h} & z \end{array} \right)$$

is an equivalence.

Proof. The map is defined by path induction, since when $h \equiv g \circ f$ the codomain is inhabited by $\text{comp}_{g, f}$. Now we again show it to be an equivalence in the general case by summing over h and noting that both types become contractible. \square

However defined, the homotopies between arrows in a Segal type behave like a 2-category up to homotopy. For instance, given $p : f =_{\text{hom}_A(x, y)} g$ and $q : g =_{\text{hom}_A(x, y)} h$, we can concatenate them as equalities to get $p \bullet q : f =_{\text{hom}_A(x, y)} h$, a “vertical” composite. We can also compose them “horizontally”:

Proposition 5.13. *Given $p : f =_{\text{hom}_A(x, y)} g$ and $q : h =_{\text{hom}_A(y, z)} k$ in a Segal type A , there is a concatenated equality $q \circ_2 p : h \circ f =_{\text{hom}_A(x, z)} k \circ g$.*

¹⁰A third case is presented by a 2-simplex in the “dual” type $\text{hom}_A^2(f, \text{id}_y; g)$, but an analogous argument will work for those; see Remark 3.1.

Proof. By path induction on p and q , defining $\text{refl}_h \circ_2 \text{refl}_f := \text{refl}_{h \circ f}$. \square

In particular, taking one of p or q to be refl but not the other, we obtain “whiskering” operations. These have another useful characterization:

Proposition 5.14. *Given $p : f =_{\text{hom}_A(x,y)} g$ and $h : \text{hom}_A(y, z)$ and $k : \text{hom}_A(w, x)$ in a Segal type A , we have*

$$\begin{aligned} \text{refl}_h \circ_2 p &= \text{ap}_{(h \circ -)}(p) \\ p \circ_2 \text{refl}_k &= \text{ap}_{(- \circ k)}(p). \end{aligned}$$

Proof. By path induction on p . \square

Of course, we have the usual middle-four interchange law:

Proposition 5.15. *We have the following equality in a Segal type whenever it makes sense:*

$$(q' \cdot p') \circ_2 (q \cdot p) = (q' \circ_2 q) \cdot (p' \circ_2 p).$$

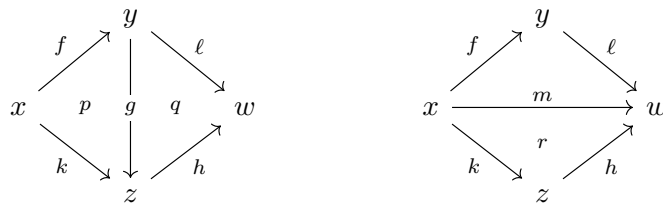
Proof. By path induction on all four equalities. \square

On the other hand, if we view homotopies as 2-simplices, then a natural way to compose them is by filling 3-dimensional horns, as in a quasicategory. We can express this in terms of whiskering and concatenation of equalities.

Proposition 5.16. *In a Segal type A , suppose given arrows f, g, h, k, ℓ, m and equalities*

$$p : g \circ f =_{\text{hom}_A(x,z)} k \qquad q : h \circ g =_{\text{hom}_A(z,w)} \ell \qquad r : h \circ k =_{\text{hom}_A(x,w)} m$$

corresponding to 2-simplices that fill out the following horn $\Lambda_2^3 \rightarrow A$:



Then the horn has a filler $\Delta^3 \rightarrow A$ in which the missing 2-face is the 2-simplex corresponding to the concatenated equality

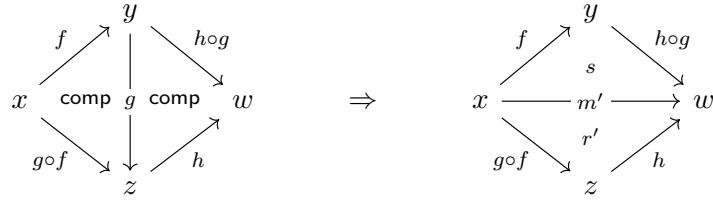
$$\ell \circ f \stackrel{q}{=} (h \circ g) \circ f = h \circ (g \circ f) \stackrel{p}{=} h \circ k \stackrel{r}{=} m. \tag{5.17}$$

where p and q are whiskered by h and f respectively.

Proof. First we do path induction on p and q , enabling us to assume $k \equiv g \circ f$ and $\ell \equiv h \circ g$. The 2-simplices corresponding to $p \equiv \text{refl}$ and $q \equiv \text{refl}$ are now $\text{comp}_{g,f}$ and $\text{comp}_{h,g}$, while (5.17) reduces to

$$(h \circ g) \circ f = h \circ (g \circ f) \stackrel{r}{=} m. \tag{5.18}$$

Now the proof of Proposition 5.9 constructed a 3-simplex of the form



By the contractibility of the type of composites for h and $g \circ f$, we have $(m', r') = (m, r)$. And the type of 3-2-horns can be decomposed as

$$(\Lambda_2^3 \rightarrow A) \simeq \sum_{\alpha: \Delta^2 \cup_{\Delta^1} \Delta^2 \rightarrow A} \langle \Delta^2 \rightarrow A \Big|_{\alpha}^{\Lambda_1^2} \rangle$$

where $\Delta^2 \cup_{\Delta^1} \Delta^2$ denotes the left-hand half of the 3-simplex drawn above, with Λ_1^2 sitting inside it as the lower two 1-simplices. Thus, the equality $(m', r') = (m, r)$ in $\langle \Delta^2 \rightarrow A \Big|_{[h, g \circ f]}^{\Lambda_1^2} \rangle$ yields an equality of 3-2-horns

$$[\mathbf{comp}_{g,f}, \mathbf{comp}_{h,g}, r'] = [\mathbf{comp}_{g,f}, \mathbf{comp}_{h,g}, r].$$

The above 3-simplex lives in $\langle \Delta^3 \rightarrow A \Big|_{[\mathbf{comp}_{g,f}, \mathbf{comp}_{h,g}, r']}^{\Lambda_2^3} \rangle$, so we can transport it across this equality to get a 3-simplex in $\langle \Delta^3 \rightarrow A \Big|_{[\mathbf{comp}_{g,f}, \mathbf{comp}_{h,g}, r]}^{\Lambda_2^3} \rangle$, which is what we wanted.

Finally, by the naturality of path transport, the missing 2-simplex face is the transport of s along the equality $m' = m$, which is equal to the concatenation $m' \stackrel{r'}{=} h \circ (g \circ f) \stackrel{r}{=} m$ of the two equalities induced by r and r' . Thus, the equality corresponding to this face is

$$(h \circ g) \circ f \stackrel{s}{=} m' \stackrel{r'}{=} h \circ (g \circ f) \stackrel{r}{=} m.$$

But the concatenation of the first two of these equalities was the *definition* of associativity $(h \circ g) \circ f = h \circ (g \circ f)$ in Proposition 5.9, so this is equal to (5.18). \square

5.5 Anodyne maps The definition of Segal type says that any 2-1-horn has a unique filler, i.e. that any extension type of the form $\langle \Delta^2 \rightarrow A \Big|_h^{\Lambda_1^2} \rangle$ is contractible. This is sufficient to imply that many other cofibrations have the same property.

Definition 5.19. An inclusion of shapes $t : I \mid \phi \vdash \psi$ is **inner anodyne** if for any Segal type A and any $h : \{t : I \mid \phi\} \rightarrow A$, the extension type $\langle \{t : I \mid \psi\} \rightarrow A \Big|_h^\phi \rangle$ is contractible.

This can only be a meta-theoretic definition, but we will not worry too much about exactly how it should be made precise; our only intent is to exhibit certain other maps as inner anodyne.

Proposition 5.20. *If $t : I \mid \phi \vdash \psi$ is inner anodyne and $s : J \mid \chi \vdash \zeta$ is any cofibration, then the pushout product*

$$\langle t, s \rangle : I \times J \mid (\phi \wedge \zeta) \vee (\psi \wedge \chi) \vdash \psi \wedge \zeta$$

is inner anodyne.

Proof. By Theorem 4.2, for any Segal type A and any h , we have

$$\begin{aligned} \langle \{I \times J \mid \psi \wedge \zeta\} \rightarrow A \Big|_h^{(\phi \wedge \zeta) \vee (\psi \wedge \chi)} \rangle \\ \simeq \langle \prod_{s:J|\zeta} \langle \{t : I \mid \psi\} \rightarrow A \Big|_{\lambda t. h(t,s)}^\phi \rangle \Big|_{\lambda s. \lambda t. h(t,s)}^\chi \rangle. \end{aligned}$$

But the latter is an extension type of a contractible family, hence contractible by relative function extensionality (Axiom 4.6). \square

Proposition 5.21. *In the cube context $\langle t_1, t_2, t_3 \rangle : \Delta^3$, the 3-1-horn and 3-2-horn inclusions*

$$\begin{aligned} (0 \equiv t_3 \leq t_2 \leq t_1) \vee (t_3 \leq t_2 \equiv t_1) \vee (t_3 \leq t_2 \leq t_1 \equiv 1) \vdash t_3 \leq t_2 \leq t_1 \\ (0 \equiv t_3 \leq t_2 \leq t_1) \vee (t_3 \equiv t_2 \leq t_1) \vee (t_3 \leq t_2 \leq t_1 \equiv 1) \vdash t_3 \leq t_2 \leq t_1 \end{aligned}$$

are inner anodyne.

Proof. This is essentially the same argument as Joyal’s lemma [19, 2.3.2.1], but written using our “interval” description of simplices rather than the “finite ordered set” version. Let $\Lambda_2^3 \rightarrow \Delta^3$ be the 3-2-horn inclusion; the 3-1 case is analogous. By Proposition 5.20, the pushout product of this inclusion with the 2-1-horn $\Lambda_1^2 \rightarrow \Delta^2$:

$$(\Lambda_2^3 \times \Delta^2) \cup_{(\Lambda_2^3 \times \Lambda_1^2)} (\Delta^3 \times \Lambda_1^2) \longrightarrow \Delta^3 \times \Delta^2$$

is inner anodyne. For brevity, let X be the domain of this pushout product. We will show that for any $h : \Lambda_2^3 \rightarrow A$, there is an $\hat{h} : X \rightarrow A$ such that $\langle \Delta^3 \rightarrow A \Big|_h^{\Lambda_2^3} \rangle$ is a retract of $\langle \Delta^3 \times \Delta^2 \rightarrow A \Big|_{\hat{h}}^X \rangle$; thus when A is Segal, the former is contractible since the latter is.

The retraction will be defined by

$$\begin{aligned} \langle \Delta^3 \times \Delta^2 \rightarrow A \Big|_{\hat{h}}^X \rangle &\rightarrow \langle \Delta^3 \rightarrow A \Big|_h^{\Lambda_2^3} \rangle \\ f &\mapsto \lambda \langle t_1, t_2, t_3 \rangle. f \langle \langle t_1, t_2, t_3 \rangle, \langle t_1, t_2 \rangle \rangle. \end{aligned}$$

This will be well-defined as long as we define \hat{h} such that

$$\hat{h} \langle \langle t_1, t_2, t_3 \rangle, \langle t_1, t_2 \rangle \rangle \equiv h \langle t_1, t_2, t_3 \rangle.$$

whenever $\langle t_1, t_2, t_3 \rangle : \Lambda_2^3$, i.e. whenever we have

$$(0 \equiv t_3 \leq t_2 \leq t_1) \vee (t_3 \equiv t_2 \leq t_1) \vee (t_3 \leq t_2 \leq t_1 \equiv 1).$$

Note that $\langle t_1, t_2, t_3 \rangle : \Lambda_2^3$ implies $\langle \langle t_1, t_2, t_3 \rangle, \langle t_1, t_2 \rangle \rangle : X$, so this condition makes sense.

If we had maximum and minimum operations \vee and \wedge in our theory, the section would be defined by

$$\begin{aligned} \langle \Delta^3 \rightarrow A \Big|_h^{\Lambda_2^3} \rangle &\rightarrow \langle \Delta^3 \times \Delta^2 \rightarrow A \Big|_{\hat{h}}^X \rangle \\ g &\mapsto \lambda \langle \langle t_1, t_2, t_3 \rangle, \langle s_1, s_2 \rangle \rangle. g \langle t_1 \vee s_1, t_2 \wedge s_2, t_3 \wedge s_2 \rangle \end{aligned}$$

and we could say that $\Lambda_2^3 \rightarrow \Delta^3$ is literally a retract of $X \rightarrow \Delta^3 \times \Delta^2$, before we even form the extension types. Instead, we have to define the section at the level of extension types only, using

rec_\vee and a sextuple case split:

$$\left\langle \Delta^3 \rightarrow A \Big|_h^{\Lambda_2^3} \right\rangle \rightarrow \left\langle \Delta^3 \times \Delta^2 \rightarrow A \Big|_{\hat{h}}^X \right\rangle$$

$$g \mapsto \lambda \langle \langle t_1, t_2, t_3 \rangle, \langle s_1, s_2 \rangle \rangle. \begin{cases} g\langle t_1, t_2, t_3 \rangle & (s_1 \leq t_1) \wedge (t_2 \leq s_2) \\ g\langle s_1, t_2, t_3 \rangle & (t_1 \leq s_1) \wedge (t_2 \leq s_2) \\ g\langle t_1, s_2, t_3 \rangle & (s_1 \leq t_1) \wedge (t_3 \leq s_2 \leq t_2) \\ g\langle s_1, s_2, t_3 \rangle & (t_1 \leq s_1) \wedge (t_3 \leq s_2 \leq t_2) \\ g\langle t_1, s_2, s_2 \rangle & (s_1 \leq t_1) \wedge (s_2 \leq t_3) \\ g\langle s_1, s_2, s_2 \rangle & (t_1 \leq s_1) \wedge (s_2 \leq t_3). \end{cases}$$

It is easy to see that this is indeed a section of the above retraction, as long as it is well-defined. We can make both the section and retraction well-defined if we define $\hat{h} : X \rightarrow A$ by the same sextuple case split from $h : \Lambda_2^3 \rightarrow A$, as long as we check that this makes sense, i.e. that if $\langle \langle t_1, t_2, t_3 \rangle, \langle s_1, s_2 \rangle \rangle : X$ then all the arguments to which the case split tries to apply g in fact live in Λ_2^3 .

It is clear that in all cases if $\langle \langle t_1, t_2, t_3 \rangle, \langle s_1, s_2 \rangle \rangle : \Delta^3 \times \Delta^2$ then the arguments of g lie in Δ^3 . The additional condition imposed by $\langle \langle t_1, t_2, t_3 \rangle, \langle s_1, s_2 \rangle \rangle : X$ is

$$(0 \equiv t_3) \vee (t_3 \equiv t_2) \vee (t_1 \equiv 1) \vee (0 \equiv s_2) \vee (s_1 \equiv 1).$$

and we have to show that each of these five cases implies

$$(0 \equiv (t_3 \wedge s_2)) \vee ((t_3 \wedge s_2) \equiv (t_2 \wedge s_2)) \vee ((t_1 \vee s_1) \equiv 1)$$

But this is easy:

- If $0 \equiv t_3$ then $0 \equiv (t_3 \wedge s_2)$.
- If $t_3 \equiv t_2$ then $(t_3 \wedge s_2) \equiv (t_2 \wedge s_2)$.
- If $t_1 \equiv 1$ then $(t_1 \vee s_1) \equiv 1$.
- If $0 \equiv s_2$ then $0 \equiv (t_3 \wedge s_2)$.
- If $s_1 \equiv 1$ then $(t_1 \vee s_1) \equiv 1$.

Technically, lacking \vee and \wedge this must be proven by splitting into $30 = 5 \times 6$ cases, but we leave that for automation in a proof assistant. \square

If we gave a formal definition of all the horns Λ_k^n (by meta-theoretic induction on n and k), then we could presumably generalize Proposition 5.21 to a theorem-schema that all the inner horn inclusions $\Lambda_k^n \rightarrow \Delta^n$ ($0 < k < n$) are inner anodyne.

Corollary 5.22. *In the situation of Proposition 5.16, if we also have $s : \ell \circ f = m$, then the type of 3-simplices with given boundary $\left\langle \Delta^3 \rightarrow A \Big|_{[p,q,r,s]}^{\partial \Delta^3} \right\rangle$ is equivalent to the type of equalities from s to (5.17).*

Proof. Proposition 5.16 gives a map from equalities to 3-simplices. As in Proposition 5.10, if we sum over s we get contractible types on both sides: one being a based path-space, the other by Proposition 5.21. Thus, the map is an equivalence. \square

6. The 2-category of Segal types

The collection of all categories is a 2-category, and similarly the collection of all $(\infty, 1)$ -categories is an $(\infty, 2)$ -category. In this section we introduce the 2-categorical structure on the collection of Segal types. Unlike in case of Segal types themselves, where we had to introduce an extra condition to characterize those types that “behave like categories”, every function between Segal types is automatically a “functor”, and every arrow between such functors is automatically a “natural transformation”. Thus, Segal types really do behave like “synthetic $(\infty, 1)$ -categories”.

6.1 Functoriality Given a function $\phi : A \rightarrow B$, we have an induced function

$$\phi_{\#} : \text{hom}_A(x, y) \rightarrow \text{hom}_B(\phi x, \phi y)$$

defined by postcomposition: $(\phi_{\#}(f))(s) = \phi(f(s))$. We usually write $\phi_{\#}$ abusively as simply ϕ . In the case where A and B are Segal types, the following observations justify our referring to any function $\phi : A \rightarrow B$ as a **functor**.

Proposition 6.1. *Any function $\phi : A \rightarrow B$ between Segal types preserves identities and composition.*

Proof. In the case of $\text{id}_x : \text{hom}_A(x, x)$, $\phi \text{id}_x : \text{hom}_B(\phi x, \phi x)$ is defined by $\phi \text{id}_x(s) \equiv \phi x$ for all $s : 2$. As $\text{id}_{\phi x}$ has the same definition, we conclude that $\text{id}_{\phi x} \equiv \phi(\text{id}_x)$, i.e. that functors preserve identities.

Similarly, ϕ gives rise to a postcomposition function

$$\phi_{\#} : \text{hom}_A^2 \left(\begin{array}{ccc} & f & y & g & \\ & \nearrow & & \searrow & \\ x & & h & & z \end{array} \right) \rightarrow \text{hom}_B^2 \left(\begin{array}{ccc} & \phi f & \phi y & \phi g & \\ & \nearrow & & \searrow & \\ \phi x & & \phi h & & \phi z \end{array} \right).$$

In particular, when A is a Segal type, we have a term

$$\phi_{\#}(\text{comp}_{g,f}) : \text{hom}_B^2 \left(\begin{array}{ccc} & \phi f & \phi y & \phi g & \\ & \nearrow & & \searrow & \\ \phi x & & \phi(g \circ f) & & \phi z \end{array} \right)$$

witnessing the fact that $\phi(g \circ f)$ is a composite of ϕg and ϕf . If B is also a Segal type, we have $(\phi g \circ \phi f, \text{comp}_{\phi g, \phi f}) = (\phi(g \circ f), \phi(\text{comp}_{g,f}))$, which implies in particular that $\phi g \circ \phi f = \phi(g \circ f)$. \square

Similarly, it can be shown that any functor between Segal types preserves all the higher-dimensional operations defined in §5.4.

6.2 Naturality

Definition 6.2. Given Segal types A and B and $f, g : A \rightarrow B$, we refer to a term $\alpha : \text{hom}_{A \rightarrow B}(f, g)$ as a **natural transformation from f to g** .

Given such an α we can apply it to any $s : 2$ to yield a term $\alpha(s) : A \rightarrow B$ so that $\alpha(0) \equiv f$ and $\alpha(1) \equiv g$. In particular, for each $a : A$, we have

$$\lambda s. \alpha(s)(a) : 2 \rightarrow B$$

which we can alternatively abstract as

$$\lambda s. \alpha(s)(a) : \text{hom}_B(fa, ga).$$

We refer to the latter as the **component of α at a** and denote it by α_a .

As expected, “a natural transformation is defined by its components”.

Proposition 6.3. *For any type B and any type or shape A , and any $f, g : A \rightarrow B$, the function*

$$\mathop{\mathrm{hom}}_{A \rightarrow B}(f, g) \rightarrow \prod_{a:A} \mathop{\mathrm{hom}}_B(fa, ga) \quad (6.4)$$

that carries a natural transformation to its components is an equivalence.

Proof. Writing out the hom-types as extension types, this function becomes

$$\left\langle \prod_{t:2} (A \rightarrow B) \Big|_{[f,g]}^2 \right\rangle \rightarrow \prod_{a:A} \left\langle \prod_{t:2} B \Big|_{[fa,ga]}^2 \right\rangle$$

and is therefore an equivalence, being an instance of Theorem 4.1 or Theorem 4.2 depending on whether A is a type or a shape. \square

We sometimes refer to the equivalence of Proposition 6.3 as **transformation extensionality**, since it is a directed analogue of function extensionality for homotopies. Since it is not just an equivalence but a judgmental isomorphism (due to the β - and η -rules for dependent function types and extension types), we generally blur the line between a natural transformation and its components.

All other structure on natural transformations is also performed componentwise. For instance, we have:

Proposition 6.5. *If B is a Segal type and A is a type or a shape, then for $\alpha : \mathop{\mathrm{hom}}_{A \rightarrow B}(f, g)$ and $\beta : \mathop{\mathrm{hom}}_{A \rightarrow B}(g, h)$ and $x : A$ we have*

$$(\beta \circ \alpha)_x = \beta_x \circ \alpha_x \quad \text{and} \quad (\mathrm{id}_f)_x = \mathrm{id}_{f(x)}.$$

Proof. By definition, we have a 2-simplex witness

$$\mathrm{comp}_{\beta, \alpha} : \mathop{\mathrm{hom}}_{A \rightarrow B}^2 \left(f \begin{array}{c} \xrightarrow{\alpha} g \xrightarrow{\beta} h \\ \xrightarrow{\beta \circ \alpha} \end{array} \right).$$

By commuting arguments and evaluating at $x : A$ this gives a 2-simplex

$$(\mathrm{comp}_{\beta, \alpha})_x : \mathop{\mathrm{hom}}_B^2 \left(f(x) \begin{array}{c} \xrightarrow{\alpha_x} g(x) \xrightarrow{\beta_x} h(x) \\ \xrightarrow{(\beta \circ \alpha)_x} \end{array} \right).$$

But we also have a 2-simplex

$$\mathrm{comp}_{\beta_x, \alpha_x} : \mathop{\mathrm{hom}}_B^2 \left(f(x) \begin{array}{c} \xrightarrow{\alpha_x} g(x) \xrightarrow{\beta_x} h(x) \\ \xrightarrow{\beta_x \circ \alpha_x} \end{array} \right)$$

so since B is Segal we have $(\beta \circ \alpha)_x = \beta_x \circ \alpha_x$. The second equality is in fact judgmental, since id_f is defined to be constant at f . \square

We now demonstrate that natural transformations really are natural.

Proposition 6.6. *For any natural transformation $\alpha : \mathop{\mathrm{hom}}_{A \rightarrow B}(f, g)$ for which B is a Segal type and $k : \mathop{\mathrm{hom}}_A(x, y)$, $\alpha_y \circ fk = gk \circ \alpha_x$.*

Proof. By the usual associativity of non-dependent function types, we can also rearrange $\alpha : \text{hom}_{A \rightarrow B}(f, g)$ into a function

$$\alpha \equiv \lambda a. \lambda s. \alpha(s)(a) : A \rightarrow B^2.$$

Now the functoriality of the extension type construction yields a map

$$\alpha_{\#} : \text{hom}_A(x, y) \rightarrow \text{hom}_{B^2}(\alpha_x, \alpha_y).$$

Given $k : \text{hom}_A(x, y)$, the term $\alpha_{\#}(k)$ is a function from 2 to B^2 , and evaluating it at $0, 1 : 2$ we get $\alpha_{\#}(k)(0) \equiv \alpha_x : \text{hom}_B(fx, gx)$ and $\alpha_{\#}(k)(1) \equiv \alpha_y : \text{hom}_B(fy, gy)$. We can also postcompose $\alpha_{\#}(k) : 2 \rightarrow B^2$ with the evaluation functions $\text{ev}_0, \text{ev}_1 : B^2 \rightarrow B$ to yield $fk : \text{hom}_B(fx, fy)$ and $gk : \text{hom}_B(gx, gy)$.

Put differently, $\alpha_{\#}(k)$ can be regarded (technically, by application and re-abstraction) as a function $2 \rightarrow B^2$, and thereby uncurried to obtain a function $2 \times 2 \rightarrow B$, and the preceding paragraph identifies the values of this function on four of the nondegenerate 1-simplices in 2×2 . If we call its value on the fifth ‘‘diagonal’’ 1-simplex α_k

$$\begin{array}{ccc} fx & \xrightarrow{\alpha_x} & gx \\ fk \downarrow & \searrow \alpha_k & \downarrow gk \\ fy & \xrightarrow{\alpha_y} & gy \end{array}$$

then the two 2-simplices contained in 2×2 yield witnesses that $\alpha_y \circ fk = \alpha_k$ and $gk \circ \alpha_x = \alpha_k$, hence $\alpha_y \circ fk = gk \circ \alpha_x$. This demonstrates that α really is natural. \square

6.3 Horizontal composition

$$f, g : A \rightarrow B \qquad j, k : B \rightarrow C \qquad \alpha : \text{hom}_{A \rightarrow B}(f, g) \qquad \beta : \text{hom}_{B \rightarrow C}(j, k)$$

where C at least is a Segal type, we can define a **horizontal composite** natural transformation

$$\beta * \alpha : \text{hom}_{C \rightarrow A}(jf, kg)$$

as follows. Define $\beta * \alpha$ to be

$$\lambda a. \beta_{\alpha_a} : A \rightarrow C^2.$$

That is, the component $(\beta * \alpha)_a : \text{hom}_C(jfa, kga)$ is defined to be the ‘‘diagonal’’ 1-simplex of the term obtained by applying the map

$$\beta_{\#} : \text{hom}_B(fa, ga) \rightarrow \text{hom}_{C^2}(\beta_{fa}, \beta_{ga})$$

to $\alpha_a : \text{hom}_B(fa, ga)$.

$$\begin{array}{ccc} jfa & \xrightarrow{\beta_{fa}} & kfa \\ j\alpha_a \downarrow & \searrow \beta_{\alpha_a} & \downarrow k\alpha_a \\ jga & \xrightarrow{\beta_{ga}} & kga \end{array}$$

The square $2 \times 2 \rightarrow C$ witnesses the ‘‘Gray interchanger’’, a homotopy between the two ways to define a ‘‘horizontal composite’’ in terms of whiskering.

7. Discrete types

In this section we will show that any type A that satisfies a natural “discreteness” condition is a Segal type. In a discrete type, the hom-types of Definition 5.1 are equivalent to its identity types, so in these types the Segal structure can be thought of as another incarnation of the weak ω -groupoid structure possessed by any type (see [30, 17]). For more on the “groupoid” interpretation of discrete types, see Proposition 10.10.

Definition 7.1. For any type A , there is a map

$$\text{idtoarr} : \prod_{x,y:A} (x =_A y) \rightarrow \text{hom}_A(x, y)$$

defined by path induction and the equation $\text{idtoarr}_{x,x}(\text{refl}_x) := \text{id}_x$. We say that A is a **discrete type** if $\text{idtoarr}_{x,y}$ is an equivalence for all $x, y : A$.

Proposition 7.2. *If $A : X \rightarrow \mathcal{U}$ is a family of discrete types, then $\prod_{x:X} A(x)$ is also discrete.*

Proof. Let $f, g : \prod_{x:X} A(x)$. By function extensionality and Theorem 4.1,

$$(f = g) \simeq \prod_{x:X} (f(x) = g(x))$$

$$\text{hom}_{\prod_{x:X} A(x)}(f, g) \simeq \prod_{x:X} \text{hom}_A(f(x), g(x)).$$

Thus, applying $\prod_{x:X}$ to the fiberwise equivalence $(f(x) = g(x)) \simeq \text{hom}_A(f(x), g(x))$ arising from the discreteness of A , we obtain the result. \square

Proposition 7.3. *If A is a discrete type then A is a Segal type.*

Proof. By Proposition 7.2, the type A^2 is also discrete; thus we know that for every $f : \text{hom}_A(x, y)$ and $g : \text{hom}_A(z, w)$, the map

$$\text{idtoarr} : (f =_{A^2} g) \rightarrow \text{hom}_{A^2}(f, g)$$

is an equivalence. Since the type of arrows A^2 in a Segal type A is equivalent to the dependent sum $\sum_{x:A} \sum_{y:A} \text{hom}_A(x, y)$, its identity types are characterized as dependent sums as well by [29, Theorem 2.7.2], and Theorem 4.3 similarly characterizes its hom-types. Thus we have an equivalence

$$\left(\sum_{e_1:x=z} \sum_{e_2:y=w} f =_{\text{hom}_A}^{(e_1, e_2)} g \right) \simeq \sum_{h:\text{hom}_A(x,z)} \sum_{k:\text{hom}_A(y,w)} \left\langle \Delta^1 \times \Delta^1 \rightarrow A \Big|_{[h,f,k,g]}^{\partial(\Delta^1 \times \Delta^1)} \right\rangle$$

where the right-hand extension type is the type of extensions from the square boundary

$$\begin{array}{ccc} x & \xrightarrow{h} & z \\ f \downarrow & & \downarrow g \\ y & \xrightarrow{k} & w \end{array}$$

into a diagram $\Delta^1 \times \Delta^1 \rightarrow A$. This equivalence projects onto the equivalence

$$\text{idtoarr} : (x = z) \times (y = w) \rightarrow \text{hom}_A(x, z) \times \text{hom}_A(y, w),$$

inducing an equivalence on the fibers over any pair of terms. Specializing to the case of $(\text{refl}_x, \text{refl}_y) : (x = x) \times (y = y)$ we see that the types

$$(f =_{\text{hom}_A(x,y)} g) \simeq \left\langle \Delta^1 \times \Delta^1 \rightarrow A \Big|_{[\text{id}_x, f, \text{id}_y, g]}^{\partial(\Delta^1 \times \Delta^1)} \right\rangle$$

are equivalent. Hence, there is an equivalence

$$\sum_{g:\text{hom}_A(x,y)} (f =_{\text{hom}_A(x,y)} g) \simeq \sum_{g:\text{hom}_A(x,y)} \left\langle \Delta^1 \times \Delta^1 \rightarrow A \Big|_{[\text{id}_x, f, \text{id}_y, g]}^{\partial(\Delta^1 \times \Delta^1)} \right\rangle,$$

and since the left-hand type is a based path space, both types are contractible.

Applying Theorem 4.4, the right-hand type is equivalent to a single extension type

$$\left\langle \Delta^1 \times \Delta^1 \rightarrow A \Big|_{[\text{id}_x, f, \text{id}_y]}^d \right\rangle \tag{7.4}$$

where d is the ‘‘cubical horn’’

$$\left(\begin{array}{ccc} \cdot & \xrightarrow{\text{id}_x} & \cdot \\ f \downarrow & & \downarrow \\ \cdot & \xrightarrow{\quad} & \cdot \end{array} \right) \longrightarrow \left(\begin{array}{ccc} \cdot & \xrightarrow{\text{id}_x} & \cdot \\ f \downarrow & \searrow & \downarrow \\ \cdot & \xrightarrow{\quad} & \cdot \end{array} \right)$$

Now to show that A is a Segal type, we must show that for all $x, y, w : A$ and $f : \text{hom}_A(x, y)$ and $k : \text{hom}_A(y, w)$, the type

$$\sum_{\ell:\text{hom}_A(x,w)} \text{hom}_A^2 \left(\begin{array}{ccc} & f & y & k \\ & \nearrow & & \searrow \\ x & & \ell & & w \end{array} \right)$$

is contractible. Since A is discrete, the hom types are equivalent to identity types, and so by path induction we may reduce to the case $y \equiv w$ and $k \equiv \text{id}_y$. In this case we have

$$\left(\sum_{\ell:\text{hom}_A(x,y)} \text{hom}_A^2 \left(\begin{array}{ccc} & f & y & \text{id}_y \\ & \nearrow & & \searrow \\ x & & \ell & & y \end{array} \right) \right) \simeq \left\langle \Delta^2 \rightarrow A \Big|_{[f, \text{id}_y]}^{\Lambda_1^2} \right\rangle.$$

To show that this type is contractible, we observe that it is a retract of the type (7.4), using the construction of Proposition 5.8 to construct the inclusion of a 2-simplex to a diagram of shape $\Delta^1 \times \Delta^1$ in which one of the new edges is an identity. \square

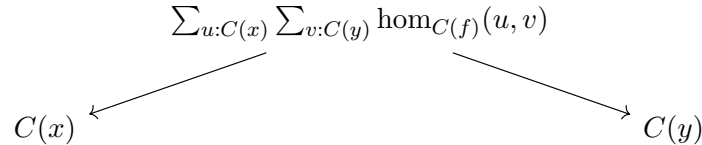
Remark 7.5. The $(\infty, 1)$ -topos of simplicial ∞ -groupoids, which is presented by our motivating model of bisimplicial sets, is a *cohesive* $(\infty, 1)$ -topos in the sense of [26], i.e. its global sections functor to ∞ -groupoids has both a right adjoint (‘‘codiscrete objects’’) and a left adjoint (‘‘discrete objects’’) that has a further product-preserving left adjoint (see also [15]). In this model, the discrete objects defined above coincide with those in the image of the ‘‘discrete objects’’ functor. It would thus be natural to enhance our type theory with modalities representing the discrete reflection, discrete coreflection, and codiscrete coreflection, as in [28]. The discrete reflection, in particular, should be constructible by ‘‘localizing at 2 ’’.

8. Covariantly functorial type families

Let $C : A \rightarrow \mathcal{U}$ be a type family. Given $x, y : A$ and $f : \text{hom}_A(x, y)$ and $u : C(x)$ and $v : C(y)$, we define the **dependent hom-type** from u to v over f to be

$$\text{hom}_{C(f)}(u, v) := \left\langle \prod_{s:2} C(f(s)) \Big|_{[u,v]}^{\partial\Delta^1} \right\rangle. \tag{8.1}$$

Intuitively, $\text{hom}_{C(f)}(u, v)$ is the type of arrows from u to v in the total space of C that lie over f . In particular, we see that C associates to every arrow $f : \text{hom}_A(x, y)$ a *span*



In general, the types $C(x)$ do not depend *functorially* on $x : A$ in the usual sense, so C cannot be regarded as a functor from A to a category of groupoids or categories. One might hope that it could be regarded as a functor to a category whose morphisms are spans, but this also fails; if in the context of

$$\begin{array}{ccccc} u : C(x) & v : C(y) & w : C(z) & k : \text{hom}_{C(f)}(u, v) & m : \text{hom}_{C(g)}(v, w) \\ \\ n : \text{hom}_{C(h)}(u, w) & & & t : \text{hom}_A^2 \left(\begin{array}{ccc} & f & y & g \\ x & \nearrow & & \searrow \\ & h & & z \end{array} \right) & \end{array}$$

we define similarly

$$\text{hom}_{C(t)}^2 \left(\begin{array}{ccc} & k & v & m \\ u & \nearrow & & \searrow \\ & n & & w \end{array} \right) := \left\langle \prod_{s:\Delta^2} C(t(s)) \Big|_{[k,m,n]}^{\partial\Delta^2} \right\rangle$$

we see that when $h = g \circ f$, the span $C(h)$ is not necessarily the composite of the spans $C(f)$ and $C(g)$, but is only related to them by a “higher span”.

We will say that C is *covariant* if each $C(x)$ is a groupoid, i.e. discrete in the sense of Definition 7.1, and moreover all of these spans are suitably representable, so that these ∞ -groupoids do depend functorially on A .¹¹ Fortunately, as with Segal types, it turns out to be sufficient to ask for one contractibility condition, which we introduce presently; in Proposition 8.18, we see that this condition implies that the fibers of a covariant fibration are necessarily discrete. We prove that the total space of a covariant family over a Segal type is itself a Segal type. We show that any fiberwise map between covariant type families induces a “natural transformation,” commuting with the functorial actions of the arrows in the base type. Then we turn our attention to the question of multivariable functoriality.

The prototypical example of a covariant family is the “representable” type family associated to a term $a : A$ in a Segal type. In §9 we will state and prove versions of the Yoneda lemma involving this notion of representable family.

8.1 Covariant fibrations

See Remark A.27 for a semantic justification of the following definition in the bisimplicial set model.

¹¹One can also consider more general *cocartesian* dependent types where the fibers are *categories* (i.e. Segal or Rezk types) depending functorially, but we leave those for later work.

Definition 8.2. We say that a type family $C : A \rightarrow \mathcal{U}$ is **covariant** if for every $f : \text{hom}_A(x, y)$ and $u : C(x)$, the type

$$\sum_{v:C(y)} \text{hom}_{C(f)}(u, v)$$

is contractible.

Dually, C is **contravariant** if for every $f : \text{hom}_A(x, y)$ and $v : C(y)$, the type

$$\sum_{u:C(x)} \text{hom}_{C(f)}(u, v)$$

is contractible; see Remark 3.1. Often we will assume that A is a Segal type.

Remark 8.3. Note that the condition that characterizes a covariant fibration is stable under substitution (i.e. precomposition or reindexing). That is, if $g : B \rightarrow A$ is a function and $C : A \rightarrow \mathcal{U}$ is a covariant type family, then $\lambda b. C(g(b)) : B \rightarrow \mathcal{U}$ is also covariant.

As for Segal types, we can reformulate this using Theorem 4.4:

Proposition 8.4. *A type family $C : A \rightarrow \mathcal{U}$ is covariant if and only if for all $f : \text{hom}_A(x, y)$ and $u : C(x)$ there is a unique lifting of f that starts at u ; i.e. the type $\langle \prod_{t:2} C(f(t)) \Big|_u^0 \rangle$ is contractible.*

Proof. By Definition 5.1 and Theorem 4.4

$$\begin{aligned} \sum_{v:C(y)} \text{hom}_{C(f)}(u, v) &\equiv \sum_{v:C(y)} \langle \prod_{t:2} C(f(t)) \Big|_{[u,v]}^{0v1} \rangle \\ &\simeq \langle \prod_{t:2} C(f(t)) \Big|_u^0 \rangle. \end{aligned}$$

Thus, Definition 8.2 asserts that C is covariant if and only if the type of extensions of u over f is contractible. \square

On the other hand, for a more global view of covariance, let us write $\tilde{C} := \sum_{z:A} C(z)$ and denote the projection by $\pi : \tilde{C} \rightarrow A$.

Theorem 8.5. *A type family $C : A \rightarrow \mathcal{U}$ is covariant if and only if the square*

$$\begin{array}{ccc} \tilde{C}^2 & \xrightarrow{\pi^2} & A^2 \\ \text{ev}_0 \downarrow & & \downarrow \text{ev}_0 \\ \tilde{C} & \xrightarrow{\pi} & A \end{array}$$

is a (homotopy) pullback.

Proof. For each $(x, u) : \tilde{C}$, we have a projection map

$$\left(\sum_{(y,v):\tilde{C}} \left(\sum_{f:\text{hom}_A(x,y)} \text{hom}_{C(f)}(u, v) \right) \right) \rightarrow \left(\sum_{y:A} \text{hom}_A(x, y) \right) \quad (8.6)$$

whose fibers are the types $\sum_{v:C(y)} \text{hom}_{C(f)}(u, v)$. Thus, Definition 8.2 is equivalent to the condition that this projection is an equivalence. If we substitute the equivalence

$$\text{hom}_{\tilde{C}}((x, u), (y, v)) \simeq \left(\sum_{f:\text{hom}_A(x,y)} \text{hom}_{C(f)}(u, v) \right)$$

established by Theorem 4.3 into (8.6), and write $p = (x, u)$ and $q = (y, v)$, it becomes

$$(\pi, \pi_{\#}) : \left(\sum_{q:\tilde{C}} \text{hom}_{\tilde{C}}(p, q) \right) \rightarrow \left(\sum_{y:A} \text{hom}_A(\pi p, y) \right). \tag{8.7}$$

Thus, Definition 8.2 is equivalent to saying that (8.7) is an equivalence for each $p : \tilde{C}$. And since a fiberwise map is a fiberwise equivalence if and only if it induces an equivalence on total spaces, this is equivalent to asking that

$$(\pi, \pi, \pi_{\#}) : \left(\sum_{p:\tilde{C}} \sum_{q:\tilde{C}} \text{hom}_{\tilde{C}}(p, q) \right) \rightarrow \left(\sum_{p:\tilde{C}} \sum_{y:A} \text{hom}_A(\pi p, y) \right)$$

is an equivalence. Finally, since $\sum_{p:\tilde{C}} \sum_{q:\tilde{C}} \text{hom}_{\tilde{C}}(p, q)$ is equivalent to \tilde{C}^2 by Theorem 4.4, and similarly

$$\begin{aligned} \left(\sum_{p:\tilde{C}} \sum_{y:A} \text{hom}_A(\pi p, y) \right) &\simeq \sum_{p:\tilde{C}} \sum_{x:A} \sum_{e:\pi p=x} \sum_{y:A} \text{hom}_A(\pi p, y) \\ &\simeq \sum_{p:\tilde{C}} \sum_{x:A} \sum_{y:A} \sum_{f:\text{hom}_A(\pi p, y)} (\pi p = x) \\ &\simeq \sum_{p:\tilde{C}} \sum_{f:A^2} (\pi p = \text{ev}_0(x)) \\ &\equiv \tilde{C} \times_A A^2 \end{aligned}$$

this is equivalent to saying that the square in the statement is a (homotopy) pullback. □

Theorem 8.8. *If A is a Segal type and $C : A \rightarrow \mathcal{U}$ is covariant, then $\tilde{C} := \sum_{z:A} C(z)$ is also a Segal type.*

We will give two proofs, or rather two versions of the same proof; one in type-theoretic language and one in category-theoretic language.

Type-theoretic proof. By Theorem 4.3, we have

$$(\Lambda_1^2 \rightarrow \tilde{C}) \simeq \sum_{\phi:\Lambda_1^2 \rightarrow A} \prod_{t:\Lambda_1^2} C(\phi(t))$$

We want to show that the type of extensions of any (ϕ, ψ) in this type to Δ^2 is contractible. This type of extensions is similarly equivalent to

$$\sum_{\mu:\langle \Delta^2 \rightarrow A \mid_{\phi}^{\Lambda_1^2} \rangle} \left\langle \prod_{t:\Delta^2} C(\mu(t)) \Big|_{\psi}^{\Lambda_1^2} \right\rangle.$$

Since A is Segal, the base of this dependent sum is contractible, so the sum itself is equivalent to

$$\left\langle \prod_{(t,s):\Delta^2} C(\text{comp}_{g,f}\langle t, s \rangle) \Big|_{\psi}^{\Lambda_1^2} \right\rangle$$

where $f : \text{hom}_A(x, y)$ and $g : \text{hom}_A(y, z)$ are the arrows making up ϕ . Now since C is covariant, $\left\langle \prod_{s:2} C(g(s)) \Big|_{\psi(1,0)}^0 \right\rangle$ is contractible, so it suffices to show that

$$\left\langle \prod_{(t,s):\Delta^2} C(\text{comp}_{g,f}\langle t, s \rangle) \Big|_{\bar{f}}^{s=0} \right\rangle$$

is contractible, where $\bar{f} : \prod_{t:2} C(f(t))$ is what is left of ψ . However, given any ν in this type, we can extend it to 2×2 with rec_ν , sending $\langle t, s \rangle$ to $\nu(t, s)$ if $s \leq t$ and $\nu(t, t)$ if $s \geq t$; cf. the proof of Proposition 3.6. Thus, this type is a retract of

$$\left\langle \prod_{\langle t,s \rangle:2 \times 2} C(c(t, s)) \Big|_{\bar{f}}^{s \equiv 0} \right\rangle$$

where $c : 2 \times 2 \rightarrow A$ is a similar extension of $\text{comp}_{g,f}$; and this is equivalent to

$$\prod_{t:2} \left\langle \prod_{s:2} C(c(t, s)) \Big|_{\bar{f}(t)}^{s \equiv 0} \right\rangle$$

and hence contractible, since it is a product of types that are each contractible by covariance of C . \square

Categorical proof. It suffices to show that $\tilde{C}^{\Delta^2} \rightarrow \tilde{C}^{\Lambda_1^2}$ is an equivalence. Consider the following pair of squares:

$$\begin{array}{ccccc} \tilde{C}^{\Delta^2} & \longrightarrow & \tilde{C}^{\Lambda_1^2} & \xrightarrow{d^2} & \tilde{C}^2 \\ \downarrow & & \downarrow & & \downarrow \\ A^{\Delta^2} & \longrightarrow & A^{\Lambda_1^2} & \xrightarrow{d^2} & A^2 \end{array} \quad (8.9)$$

We have $A^{\Lambda_1^2} \simeq \sum_{p:A^2} \sum_{z:A} \text{hom}_A(p(1), z)$ and similarly for \tilde{C} , so covariance of π implies that the right-hand square is a pullback. Since A is a Segal type, the bottom-left arrow $A^{\Delta^2} \rightarrow A^{\Lambda_1^2}$ is an equivalence; thus it will suffice to show that the outer rectangle is also a pullback.

Now we also have $A^{2 \times 2} \simeq \sum_{p:A^2} \prod_{s:2} \sum_{z:A} \text{hom}_A(p(s), z)$ and similarly for \tilde{C} , so covariance of π also implies that the square below is a pullback:

$$\begin{array}{ccc} \tilde{C}^{2 \times 2} & \xrightarrow{(\text{Id}_2, [0])} & \tilde{C}^2 \\ \downarrow & & \downarrow \\ A^{2 \times 2} & \xrightarrow{(\text{Id}_2, [0])} & A^2 \end{array} \quad (8.10)$$

Now we recall again that A^{Δ^2} is a retract of $A^{2 \times 2}$. Since this retraction is natural, the outer rectangle in (8.9) is a retract of the square (8.10); hence it is also a pullback. \square

Remark 8.11 (dependent composition). Since $\sum_{z:A} C(z)$ is a Segal type, we can compose arrows in it. However, it is often more useful to compose “dependent arrows” in the following sense. Given $f : \text{hom}_A(x, y)$ and $g : \text{hom}_A(y, z)$ and also $k : \text{hom}_{C(f)}(u, v)$ and $m : \text{hom}_{C(g)}(v, w)$, we showed that for any $h : \text{hom}_A(x, z)$ and $t : \text{hom}_A^2(f, g; h)$ the type

$$\sum_{n:\text{hom}_{C(h)}(u, w)} \text{hom}_{C(t)}^2 \left(u \begin{array}{c} \xrightarrow{k} v \\ \xrightarrow[n]{} w \end{array} \xrightarrow{m} \right) \quad (8.12)$$

is contractible. In the case when $h := g \circ f$ and $t := \text{comp}_{g,f}$, we will write the specified inhabitant of (8.12) as $(m \circ k, \text{comp}_{m,k})$. Note that $m \circ k$ is, by definition, an arrow “over” $g \circ f$, and similarly $\text{comp}_{m,k}$ is a 2-simplex over $\text{comp}_{g,f}$.

We now show that each element of a Segal type gives rise to a covariant “representable” type family and in fact the covariance condition on this type family characterizes Segal types.

Proposition 8.13. *Let A be a type and fix $a : A$. Then the type family*

$$\lambda x. \text{hom}_A(a, x) : A \rightarrow \mathcal{U}$$

is covariant if and only if A is a Segal type.

Proof. The condition of Definition 8.2 asserts that for each $b, c : A$, $f : \text{hom}_A(a, b)$, and $g : \text{hom}_A(b, c)$, the type

$$\sum_{h : \text{hom}_A(a, c)} \left\langle \prod_{s:2} \text{hom}_A(a, g(s)) \Big|_{[f, h]}^{\partial \Delta^1} \right\rangle$$

is contractible. Applying Theorem 4.4, this is easily seen to be equivalent to

$$\left\langle 2 \times 2 \rightarrow A \Big|_{[\text{id}_a, f, g]}^d \right\rangle$$

where d is the “cubical horn”

$$\left(\begin{array}{ccc} \cdot & \xrightarrow{\text{id}_a} & \cdot \\ f \downarrow & & \cdot \\ \cdot & \xrightarrow{g} & \cdot \end{array} \right) \longrightarrow \left(\begin{array}{ccc} \cdot & \xrightarrow{\text{id}_a} & \cdot \\ f \downarrow & \searrow & \cdot \\ \cdot & \xrightarrow{g} & \cdot \end{array} \right)$$

But since 2×2 is the pushout of two copies of Δ^2 over their diagonal faces, our type is now also equivalent to

$$\sum_{k : \text{hom}_A(a, c)} \left(\text{hom}_A^2 \left(\begin{array}{ccc} & f & b & g \\ a & \xleftarrow{\quad} & & \xrightarrow{\quad} \\ & k & & c \end{array} \right) \times \sum_{h : \text{hom}_A(a, c)} \text{hom}_A^2 \left(\begin{array}{ccc} & \text{id}_a & a & h \\ a & \xleftarrow{\quad} & & \xrightarrow{\quad} \\ & k & & c \end{array} \right) \right)$$

Now by Proposition 5.10, we have

$$\left(\sum_{h : \text{hom}_A(a, c)} \text{hom}_A^2 \left(\begin{array}{ccc} & \text{id}_a & a & h \\ a & \xleftarrow{\quad} & & \xrightarrow{\quad} \\ & k & & c \end{array} \right) \right) \simeq \sum_{h : \text{hom}_A(a, c)} (h = k),$$

which is contractible. Thus, it remains to consider

$$\sum_{k : \text{hom}_A(a, c)} \text{hom}_A^2 \left(\begin{array}{ccc} & f & b & g \\ a & \xleftarrow{\quad} & & \xrightarrow{\quad} \\ & k & & c \end{array} \right)$$

which is contractible if and only if A is a Segal type. □

Of course, by duality, $\lambda x. \text{hom}_A(x, a)$ is contravariant.

8.2 Functoriality If $C : A \rightarrow \mathcal{U}$ is covariant, then the arrows of A act on C in the following way. Given $f : \text{hom}_A(x, y)$ and $u : C(x)$, by assumption $\sum_{v : C(y)} \text{hom}_{C(f)}(u, v)$ is contractible, and in particular inhabited. We write its specified inhabitant as $(f_*u, \text{trans}_{f, u})$.

Example 8.14. In the case of the covariant representable $\lambda x. \text{hom}_A(a, x) : A \rightarrow \mathcal{U}$, suppose $e : \text{hom}_A(a, x)$ and $f : \text{hom}_A(x, y)$. Then the proof of Proposition 8.13 shows that $f_*e = f \circ e$.

We have an analogue of Proposition 5.10. Note that this is also a directed version of the usual characterization of “dependent paths” in homotopy type theory as paths whose domain is a transport [29, (6.2.2)].

Lemma 8.15. *If $C : A \rightarrow \mathcal{U}$ is covariant and $f : \text{hom}_A(x, y)$ and $u : C(x)$ and $v : C(y)$, then*

$$\text{hom}_{C(f)}(u, v) \simeq (f_*u =_{C(y)} v)$$

Proof. Given $g : \text{hom}_{C(f)}(u, v)$, we have $(v, g) : \sum_{w:C(y)} \text{hom}_{C(f)}(u, w)$, hence $(f_*u, \text{trans}_{f,u}) = (v, g)$ by contractibility and so $f_*u = v$. This gives a map from left to right. To show that it is an equivalence, we observe that the induced map on total spaces

$$\left(\sum_{v:C(y)} \text{hom}_{C(f)}(u, v) \right) \rightarrow \left(\sum_{v:C(y)} (f_*u = v) \right)$$

is an equivalence, since both types are contractible. \square

We now argue that the operation that takes $f : \text{hom}_A(x, y)$ and $u : C(x)$ and produces $f_*u : C(y)$ is “functorial” in the expected sense:

Proposition 8.16. *Suppose A is a Segal type and $C : A \rightarrow \mathcal{U}$ is covariant. Then given $f : \text{hom}_A(x, y)$, $g : \text{hom}_A(y, z)$, and $u : C(x)$, we have*

$$g_*(f_*u) = (gf)_*u \quad \text{and} \quad (\text{id}_x)_*u = u.$$

Proof. Given $f : \text{hom}_A(x, y)$ and $u : C(x)$, we have

$$(f_*u, \text{trans}_{f,u}) : \sum_{v:C(y)} \text{hom}_{C(f)}(u, v).$$

Now suppose given also $g : \text{hom}_A(y, z)$. Then we have also

$$(g_*(f_*u), \text{trans}_{g,f_*u}) : \sum_{w:C(z)} \text{hom}_{C(g)}(f_*u, w)$$

and

$$((gf)_*u, \text{trans}_{gf,u}) : \sum_{w:C(z)} \text{hom}_{C(gf)}(u, w).$$

On the other hand, the dependent composition $\text{trans}_{g,f_*u} \circ \text{trans}_{f,u}$ discussed in Remark 8.11 lies in the type $\text{hom}_{C(gf)}(u, g_*(f_*u))$, and so we have

$$(g_*(f_*u), \text{trans}_{g,f_*u} \circ \text{trans}_{f,u}) : \sum_{w:C(z)} \text{hom}_{C(gf)}(u, w).$$

Thus, since this type is contractible, we have $g_*(f_*u) = (gf)_*u$.

The case of identities is even easier. Given $u : C(x)$, by definition we have

$$((\text{id}_x)_*u, \text{trans}_{\text{id}_x,u}) : \sum_{v:C(x)} \text{hom}_{C(\text{id}_x)}(u, v)$$

but we also have a dependent identity arrow $\text{id}_u : \text{hom}_{C(\text{id}_x)}(u, u)$ and so

$$(u, \text{id}_u) : \sum_{v:C(x)} \text{hom}_{C(\text{id}_x)}(u, v)$$

By contractibility, therefore, $(\text{id}_x)_*u = u$. \square

8.3 Naturality Any fiberwise map between two covariant fibrations $C, D : A \rightarrow \mathcal{U}$ defines a “natural transformation”, commuting with the functorial action of Proposition 8.16:

Proposition 8.17. *Suppose given two covariant $C, D : A \rightarrow \mathcal{U}$ and a fiberwise map $\phi : \prod_{x:A} C(x) \rightarrow D(x)$. Then for any $f : \text{hom}_A(x, y)$ and $u : C(x)$,*

$$\phi_y(f_*u) = f_*(\phi_x(u)).$$

Proof. We can apply ϕ to $\text{trans}_{f,u}$ to obtain

$$(\phi_y(f_*u), \lambda t. \phi_{f(t)}(\text{trans}_{f,u}(t))) : \sum_{v:D(\phi_y(u))} \text{hom}_{D(f)}(\phi_x(u), v).$$

But of course we also have

$$(f_*(\phi_x(u)), \text{trans}_{f,\phi_x(u)}) : \sum_{v:D(\phi_y(u))} \text{hom}_{D(f)}(\phi_x(u), v)$$

so by contractibility $\phi_y(f_*u) = f_*(\phi_x(u))$. □

8.4 Discrete fibers The fibers of a covariant fibration over a Segal type are discrete types.

Proposition 8.18. *If A is a Segal type and $C : A \rightarrow \mathcal{U}$ is a covariant type family, then for each $x : A$, the type $C(x)$ is discrete.*

Proof. We must show that $\text{idtoarr} : (u = v) \rightarrow \text{hom}_{C(x)}(u, v)$ is an equivalence for all $u, v : C(x)$. It suffices to show that the induced map on total spaces

$$\left(\sum_{v:C(x)} (u = v) \right) \rightarrow \left(\sum_{v:C(x)} \text{hom}_{C(x)}(u, v) \right)$$

is an equivalence. But its domain is contractible since it is a based path space, and its codomain is contractible by covariance of C applied to $\text{id}_x : \text{hom}_A(x, x)$. □

Corollary 8.19. *If A is a Segal type and $x, y : A$, then $\text{hom}_A(x, y)$ is discrete.*

Proof. By Propositions 8.13 and 8.18. □

With Proposition 7.2 it follows that various other types are discrete. For instance, if $B, C : A \rightarrow \mathcal{U}$ are covariant, then the type $\prod_{a:A} (B(a) \rightarrow C(a))$ of “natural transformations” from B to C is discrete. We also have:

Corollary 8.20. *If A is discrete then so is $x =_A y$ for any $x, y : A$.*

Proof. Since A is discrete, it is Segal, hence $\text{hom}_A(x, y)$ is discrete. But since A is discrete, $x =_A y$ is equivalent to $\text{hom}_A(x, y)$ and hence also discrete. □

8.5 Multivariable covariance We say that a type family dependent on multiple types is **covariant** if it is covariant in the ordinary sense when regarded as dependent on the Σ -type that collects all its arguments. For instance, $C : A \rightarrow B \rightarrow \mathcal{U}$ is covariant if its uncurried version $C' : A \times B \rightarrow \mathcal{U}$ is covariant. In this case we have:

Proposition 8.21. *$C : A \rightarrow B \rightarrow \mathcal{U}$ is covariant if and only if $C(a, -)$ is covariant for each $a : A$ and $C(-, b)$ is covariant for each $b : B$.*

Proof. “Only if” follows from Remark 8.3. For “if”, note first that by Theorem 4.3 we have $\text{hom}_{A \times B}((a, b), (a', b')) \simeq \text{hom}_A(a, a') \times \text{hom}_B(b, b')$. For $f : \text{hom}_A(a, a')$ and $g : \text{hom}_B(b, b')$, we write $(f, g) : \text{hom}_{A \times B}((a, b), (a', b'))$ by abuse of notation. By Proposition 8.4 we must show that for any $u : C(a, b)$ the type

$$\left\langle \prod_{t:2} C(f(t), g(t)) \Big|_u^0 \right\rangle. \quad (8.22)$$

is contractible. We will show that this is a retract of

$$\left\langle \prod_{(t,s):2 \times 2} C(f(t), g(s)) \Big|_u^{(0,0)} \right\rangle \quad (8.23)$$

and that (8.23) is contractible. For the latter, we rewrite (8.23) using Theorems 4.2 and 4.4 as

$$\sum_{\phi: \langle \prod_{t:2} C(f(t), b) \Big|_u^0 \rangle} \prod_{t:2} \left\langle \prod_{s:2} C(f(t), g(s)) \Big|_{\phi(t)}^0 \right\rangle$$

Now $\langle \prod_{t:2} C(f(t), b) \Big|_u^0 \rangle$ is contractible since $C(-, b)$ is covariant, with center $\text{trans}_{f,u}$. So (8.23) is equivalent to

$$\prod_{t:2} \left\langle \prod_{s:2} C(f(t), g(s)) \Big|_{(\text{trans}_{f,u})(t)}^0 \right\rangle$$

But since $C(f(t), -)$ is covariant, $\left\langle \prod_{s:2} C(f(t), g(s)) \Big|_{(\text{trans}_{f,u})(t)}^0 \right\rangle$ is contractible for any $t : 2$; thus (8.23) is contractible by relative function extensionality.

It remains to show that (8.22) is a retract of (8.23). The retraction is just evaluation on the diagonal: $\phi \mapsto \lambda t. \phi(t, t)$. For the section, suppose given $\phi : \langle \prod_{t:2} C(f(t), g(t)) \Big|_u^0 \rangle$. We want to define an element of $\prod_{(t,s)} C(f(t), g(s))$; this will be defined by gluing together a pair of 2-simplices defined for $t \leq s$ and $s \leq t$ respectively that restrict judgmentally to $\phi(t)$ on the 1-simplex $t = s$.

Recall from Proposition 3.5 that we have a connection square \mathbf{V}_g with the following faces:

$$\begin{array}{ccc} b & \xrightarrow{g} & b' \\ g \downarrow & \searrow g & \downarrow \text{id}_{b'} \\ b' & \xrightarrow{\text{id}_{b'}} & b' \end{array}$$

We define

$$g_t := \lambda s. \mathbf{V}_g(t, s) : \text{hom}_B(g(t), b').$$

Thus if $s \leq t$ then $g_t(s) \equiv g(t)$, while if $t \leq s$ then $g_t(s) \equiv g(s)$. Thus we have the covariant lifting arrow $\text{trans}_{g_t, \phi(t)} : \text{hom}_{C(f(t), g_t)}(\phi(t), (g_t)_*(\phi(t)))$ with respect to the type family $C(f(t), -)$, and evaluating it at s we have $\text{trans}_{g_t, \phi(t)}(s) : C(f(t), g_t(s))$.

Similarly, we define $f_s := \lambda t. \mathbf{V}_f(t, s)$, yielding $\text{trans}_{f_s, \phi(s)}(t) : C(f_s(t), g(s))$. Since $C(f(t), g_t(s)) \equiv C(f(t), g(s))$ for $t \leq s$ and $C(f_s(t), g(s)) \equiv C(f(t), g(s))$ for $s \leq t$, we would like to paste these together with rec_\vee to get

$$i \quad \psi(t, s) = \begin{cases} \text{trans}_{g_t, \phi(t)}(s) & t \leq s \\ \text{trans}_{f_s, \phi(s)}(t) & s \leq t \end{cases} \quad ?$$

But unfortunately we do not know that these two values agree when $s \equiv t$. We know that $\text{trans}_{g_t, \phi(t)}(0) \equiv \phi(t)$ and $\text{trans}_{f_s, \phi(s)}(0) \equiv \phi(s)$, but although g_t is constant for $0 \leq s \leq t$ it doesn't follow that $\text{trans}_{g_t, \phi(t)}$ is constant on that range, so that $\text{trans}_{g_t, \phi(t)}(t)$ might not equal $\phi(t)$. Put differently, we have

$$\lambda(t, s). \text{trans}_{g_t, \phi(t)}(s) : \left\langle \prod_{(t,s)|t \leq s} C(f(t), g(s)) \Big|_{\lambda(t,s). \text{trans}_{g_t, \phi(t)}(t)}^{t \equiv s} \right\rangle \quad (8.24)$$

whereas we need something in

$$\left\langle \prod_{(t,s)|t \leq s} C(f(t), g(s)) \Big|_{\lambda_{(t,s)} \cdot \phi(t)}^{t \equiv s} \right\rangle. \quad (8.25)$$

Consider the 2-simplex that we would like to be degenerate but isn't, $\mathbf{trans}_{g_t, \phi(t)}(s)$ for $s \leq t$. We can use \mathbf{rec}_\vee to put this together with a 2-simplex $\mathbf{trans}_{g_t, \phi(t)}(t)$ for $t \leq s$ that is degenerate:

$$h(t, s) := \begin{cases} \mathbf{trans}_{g_t, \phi(t)}(s) & s \leq t \\ \mathbf{trans}_{g_t, \phi(t)}(t) & t \leq s \end{cases}$$

Then $h(t, s) : C(f(t), g(t))$ for all t and s , since $g_t(s) \equiv g(t)$ for $s \leq t$. Thus for each t , we have an arrow $\lambda s. h(t, s) : \mathbf{hom}_{C(f(t), g(t))}(\phi(t), \mathbf{trans}_{g_t, \phi(t)}(t))$. Since Proposition 8.18 proves that $C(f(t), g(t))$ is discrete, this yields an equality $\phi(t) = \mathbf{trans}_{g_t, \phi(t)}(t)$, and thus an equality $(\lambda_{(t,s)} \cdot \phi(t)) = (\lambda_{(t,s)} \cdot \mathbf{trans}_{g_t, \phi(t)}(t))$ by relative function extensionality. Therefore, we can transport (8.24) along this equality to get an element of (8.25) as desired. We argue similarly on the opposite side to obtain $\phi(s) = \mathbf{trans}_{f_s, \phi(s)}(s)$ and another 2-simplex in

$$\left\langle \prod_{(t,s)|s \leq t} C(f(t), g(s)) \Big|_{\lambda_{(t,s)} \cdot \phi(s)}^{t \equiv s} \right\rangle$$

that we can glue with this one, giving the desired section. \square

We can also consider such families where one variable depends on another one. For instance, $C : \prod_{a:A} (B(a) \rightarrow \mathcal{U})$ is covariant if its uncurried version $C' : (\sum_{a:A} B(a)) \rightarrow \mathcal{U}$ is covariant. A fundamental example is the following.

Theorem 8.26. *Suppose $C : A \rightarrow \mathcal{U}$ is covariant. Then*

$$\lambda a. \lambda u. \lambda v. (u = v) : \prod_{a:A} (C(a) \rightarrow C(a) \rightarrow \mathcal{U})$$

is also covariant.

Proof. The family $(\lambda a. C(a) \times C(a)) : A \rightarrow \mathcal{U}$ is covariant, so an arrow in its total space is uniquely determined by an arrow $f : \mathbf{hom}_A(a, a')$ and a lift $(u, v) : C(a) \times C(a)$ of its domain. We denote the resulting uniquely determined arrow by

$$\phi_{u,v}^f : \mathbf{hom}_{\sum_{a:A} C(a) \times C(a)}((a, u, v), (a', f_*u, f_*v)).$$

By Theorem 4.3, the type of $\phi_{u,v}^f$ is equivalent to

$$\sum_{f:\mathbf{hom}_A(a,a')} \mathbf{hom}_{C(f)}(u, f_*u) \times \mathbf{hom}_{C(f)}(v, f_*v)$$

and under this equivalence $\phi_{u,v}^f$ corresponds to the triple $(f, \mathbf{trans}_{f,u}, \mathbf{trans}_{f,v})$.

Now suppose $p : u = v$; we want to show that the following type is contractible:

$$\left\langle \prod_{t:2} \mathbf{trans}_{f,u}(t) =_{C(f(t))} \mathbf{trans}_{f,v}(t) \Big|_p^0 \right\rangle.$$

By path induction, we are free to assume that v is u and that p is \mathbf{refl} . However, now by relative function extensionality we have

$$\left\langle \prod_{t:2} \mathbf{trans}_{f,u}(t) =_{C(f(t))} \mathbf{trans}_{f,u}(t) \Big|_{\mathbf{refl}_u}^0 \right\rangle \simeq \left(\mathbf{trans}_{f,u} =_{\langle \prod_{t:2} C(f(t)) \Big|_u^0 \rangle} \mathbf{trans}_{f,u} \right)$$

and the latter is contractible since it is a path space in a type that is itself contractible, since C is covariant. \square

It is also useful to identify the covariant transport in such a family.

Proposition 8.27. *With notation as in the proof of Theorem 8.26, for any equality $e : u =_{C(a)} v$ we have $(\phi_{u,v}^f)_* e = \mathbf{ap}_{f_*}(e)$.*

Proof. By path induction, we assume $v \equiv u$ and $e \equiv \mathbf{refl}$. But then $\lambda t. \mathbf{refl}_{\mathbf{trans}_{f,u}(t)}$ is a lift of $\phi_{u,u}^f$ starting at \mathbf{refl}_u and ending at \mathbf{refl}_{f_*u} , so $(\phi_{u,u}^f)_* \mathbf{refl}_u = \mathbf{refl}_{f_*u}$, which is by definition $\mathbf{ap}_{f_*}(\mathbf{refl}_u)$. \square

8.6 Two-sided discrete fibrations We now consider type families dependent on multiple types with opposite variance.

Definition 8.28. Let A and B be Segal types and let $C : A \rightarrow B \rightarrow \mathcal{U}$ be a type family. We say that C is **contravariant over A and covariant over B** if for all $a : A$ and $b : B$ the type families

$$\lambda y. C(a, y) : B \rightarrow \mathcal{U} \quad \text{and} \quad \lambda x. C(x, b) : A \rightarrow \mathcal{U}$$

respectively define a covariant family over B and a contravariant family over A .

In classical category theory, fibrations of the form of Definition 8.28 are called **two-sided discrete fibrations**. The prototypical example is given by Proposition 8.13 and its dual:

Proposition 8.29. *If A is a Segal type, then the type family*

$$\lambda x. \lambda y. \mathbf{hom}_A(x, y) : A \rightarrow A \rightarrow \mathcal{U}$$

is a two-sided discrete fibration. \square

8.7 Closure properties of covariance Mapping into a covariant family (even dependently) preserves covariance; while mapping out of a covariant family, at least into a discrete type, yields a contravariant family. The former is easy to prove, but the latter is rather trickier.

Theorem 8.30. *Let $C : A \rightarrow B \rightarrow \mathcal{U}$ be such that each $C(-, b)$ is covariant. Then $\lambda a. \prod_b C(a, b) : A \rightarrow \mathcal{U}$ is also covariant.*

Proof. By Proposition 8.4 we must show that every $f : \mathbf{hom}_A(a, a')$ has a unique lifting that starts at $g : \prod_{b:B} C(a, b)$. By Theorem 4.1, the type of such extensions, displayed below-left, is equivalent to the dependent function type displayed below-right:

$$\left\langle \prod_{t:2} \left(\prod_{b:B} C(f(t), b) \right) \Big|_g^0 \right\rangle \simeq \prod_{b:B} \left\langle \prod_{t:2} C(f(t), b) \Big|_{g(b)}^0 \right\rangle.$$

Since $C(-, b)$ is covariant, each $\left\langle \prod_{t:2} C(f(t), b) \Big|_{g(b)}^0 \right\rangle$ is contractible, hence so is the right-hand side. \square

Theorem 8.31. *Let $C : A \rightarrow \mathcal{U}$ be covariant and let Y be discrete. Then $\lambda a. (C(a) \rightarrow Y) : A \rightarrow \mathcal{U}$ is contravariant.*

Proof. Fix $f : \mathbf{hom}_A(a, a')$ and $v : C(a') \rightarrow Y$; we must show that

$$\left\langle \prod_{t:2} (C(f(t)) \rightarrow Y) \Big|_v^1 \right\rangle \tag{8.32}$$

is contractible. Recall from Proposition 3.5 that f gives rise to a square \mathbf{V}_f with the following faces:

$$\begin{array}{ccc} a & \xrightarrow{f} & a' \\ f \downarrow & \searrow f & \downarrow \text{id}_{a'} \\ a' & \xrightarrow{\text{id}_{a'}} & a' \end{array}$$

As in Proposition 8.21, we define

$$f_t := \lambda s. \mathbf{V}_f(t, s) : \text{hom}_A(f(t), a').$$

Thus if $s \leq t$ then $f_t(s) \equiv f(t)$, while if $t \leq s$ then $f_t(s) \equiv f(s)$. Then for any t and any $c : C(f(t))$ we have $(f_t)_*c : C(a')$ and hence $v((f_t)_*c) : Y$, so (8.32) is inhabited by $\lambda t. \lambda c. v((f_t)_*c)$.

It remains to show that any element of (8.32) is equal to $\lambda t. \lambda c. v((f_t)_*c)$. Thus, let $\phi : \langle \prod_{t:2} (C(f(t)) \rightarrow Y) \rangle_v^1$; by relative function extensionality it suffices to fix $t : 2$ and $c : C(f(t))$ and show $\phi(t, c) = v((f_t)_*c)$. And in fact, since Y is discrete, it suffices to define an arrow in $\text{hom}_Y(\phi(t, c), v((f_t)_*c))$.

Now we have $\text{trans}_{f_t, c} : \text{hom}_{C(f_t)}(c, (f_t)_*c)$, and thus for any $s : 2$ we have $\text{trans}_{f_t, c}(s) : C(f_t(s))$. Thus if $s \leq t$, we can write $\phi(t, \text{trans}_{f_t, c}(s))$; while we can always write $\phi(t, \text{trans}_{f_t, c}(t))$ since $f_t(t) \equiv f(t)$ for any t . Using rec_v , we may paste these 2-simplices together to define

$$k(t, s) := \begin{cases} \phi(t, \text{trans}_{f_t, c}(s)) & s \leq t \\ \phi(t, \text{trans}_{f_t, c}(t)) & t \leq s. \end{cases}$$

For each t , this provides an arrow $\lambda s. k(t, s) : \text{hom}_Y(\phi(t, c), \phi(t, \text{trans}_{f_t, c}(t)))$ since $\text{trans}_{f_t, c}(0) \equiv c$.

Similarly, using rec_v , we may paste together a pair of 2-simplices to define

$$h(t, s) := \begin{cases} \phi(t, \text{trans}_{f_t, c}(t)) & s \leq t \\ \phi(s, \text{trans}_{f_t, c}(s)) & t \leq s. \end{cases}$$

For each t , $\phi(1, \text{trans}_{f_t, c}(1)) \equiv \phi(1, (f_t)_*(c)) \equiv v((f_t)_*c)$, so this provides an arrow $\lambda s. h(t, s) : \text{hom}_Y(\phi(t, \text{trans}_{f_t, c}(t)), v((f_t)_*c))$. Thus $\phi(t, c) = \phi(t, \text{trans}_{f_t, c}(t))$ and $\phi(t, \text{trans}_{f_t, c}(t)) = v((f_t)_*c)$ since Y is discrete. \square

9. The Yoneda lemma

Let $C : A \rightarrow \mathcal{U}$ be covariant, and fix $a : A$. Then we have maps

$$\begin{aligned} \text{evid}_a^C &:= \lambda \phi. \phi(a, \text{id}_a) & : & \left(\prod_{x:A} (\text{hom}_A(a, x) \rightarrow C(x)) \right) \rightarrow C(a) \\ \text{yon}_a^C &:= \lambda u. \lambda x. \lambda f. f_*u & : & C(a) \rightarrow \left(\prod_{x:A} (\text{hom}_A(a, x) \rightarrow C(x)) \right). \end{aligned}$$

Theorem 9.1 (Yoneda lemma). *If A is a Segal type, then for any covariant $C : A \rightarrow \mathcal{U}$ and $a : A$, the maps evid_a^C and yon_a^C are inverse equivalences.*

Proof. In one direction, given $u : C(a)$ we have

$$(\lambda x. \lambda f. f_*u)(a, \text{id}_a) = (\text{id}_a)_*u = u$$

by Proposition 8.16. In the other direction, we want to compare the fiberwise map ϕ to $\lambda x. \lambda f. f_*(\phi(a, \text{id}_a))$. By function extensionality, we can evaluate both of them at some $x : A$ and $f : \text{hom}_A(a, x)$, in which case we have

$$f_*(\phi(a, \text{id}_a)) = \phi(x, f_*\text{id}_a) = \phi(x, f \circ \text{id}_a) = \phi(x, f)$$

by Proposition 8.17 and Example 8.14. \square

This is of course just the usual proof of the Yoneda lemma. However, note that we do not need to manually check the naturality of $\text{yon}_a^C(u)$; this is automatic simply by its being defined as a fiberwise map. Similarly, because its domain and codomain are both covariant in $a : A$ (the domain by Theorems 8.30 and 8.31 — or simply by the fact of being fiberwise equivalent to the codomain), the Yoneda equivalence is itself automatically natural in a . Naturality in C is not similarly automatic, but is easy to prove:

Lemma 9.2. *If A is a Segal type and $a : A$, while $C, D : A \rightarrow \mathcal{U}$ are covariant and $\psi : \prod_{x:A} (C(x) \rightarrow D(x))$, then we have*

$$\begin{aligned} \psi_a \circ \text{evid}_a^C &\equiv \text{evid}_a^D \circ (\lambda\phi. \lambda x. \lambda f. \psi_x(\phi(x, f))) \\ (\lambda\phi. \lambda x. \lambda f. \psi_x(\phi(x, f))) \circ \text{yon}_a^C &= \text{yon}_a^D \circ \psi_a \end{aligned}$$

Proof. The first is simple β -reduction: both sides equal $\lambda\phi. \psi_a(\phi(a, \text{id}_a))$. In the second, the left-hand side equals $\lambda u. \lambda x. \lambda f. \psi_x(f_*u)$ while the right-hand side equals $\lambda u. \lambda x. \lambda f. f_*\psi_a(u)$; thus the equality follows from Proposition 8.17 (and function extensionality). \square

Definition 9.3. When $a, a' : A$ are terms in a Segal type, we refer to the map

$$\text{yon}_a^{\text{hom}_A(a', -)} : \text{hom}_A(a', a) \rightarrow \left(\prod_{x:A} \text{hom}_A(a, x) \rightarrow \text{hom}_A(a', x) \right)$$

as the **Yoneda embedding**.

Remark 9.4. Because the Yoneda embedding is an equivalence, we know that any fiberwise map $\phi : \prod_{x:A} \text{hom}_A(a, x) \rightarrow \text{hom}_A(a', x)$ between covariant representables for a Segal type A is equal to a post-composition function. Namely, if $u := \text{evid}_a^{\text{hom}_A(a', -)}(\phi) : \text{hom}_A(a', a)$ then by Example 8.14,

$$\phi = \text{yon}_a^{\text{hom}_A(a', -)}(u) = \lambda x. \lambda f. f_*u = \lambda x. \lambda f. f \circ u,$$

which is to say the natural transformation ϕ is given by precomposition with the arrow $u : \text{hom}_A(a', a)$.

From a type-theoretic perspective, the Yoneda lemma is a “directed” version of the “transport” operation for identity types. This suggests a “dependently typed” generalization of the Yoneda lemma, analogous to the full induction principle for identity types. Recall from §8.5 that a type family $C : \prod_{x:A} (\text{hom}(a, x) \rightarrow \mathcal{U})$ is called covariant if its uncurried version $C : \left(\sum_{x:A} \text{hom}(a, x) \right) \rightarrow \mathcal{U}$ is covariant.

Theorem 9.5 (dependent Yoneda lemma). *If A is a Segal type, $a : A$, and $C : \prod_{x:A} (\text{hom}(a, x) \rightarrow \mathcal{U})$ is covariant, then the function*

$$\text{evid}_a^C := \lambda\phi. \phi(a, \text{id}_a) : \left(\prod_{x:A} \prod_{f:\text{hom}_A(a, x)} C(x, f) \right) \rightarrow C(a, \text{id}_a)$$

is an equivalence.

We will obtain this as a special case of a result about types with initial objects.

Definition 9.6. A point $b : B$ is **initial** if for all $x : B$ the hom-type $\text{hom}_B(b, x)$ is contractible.

Theorem 9.7. *If $b : B$ is initial and $C : B \rightarrow \mathcal{U}$ is covariant, then the function*

$$\lambda\phi. \phi(b) : \left(\prod_{x:B} C(x) \right) \rightarrow C(b)$$

is an equivalence.

Proof. Since each $\text{hom}_B(b, x)$ is contractible, it is in particular inhabited, so we have some $f : \prod_{x:B} \text{hom}_B(b, x)$. Moreover, since $\text{hom}_B(b, b)$ is contractible, we have $f_b = \text{id}_b$.

Now for an inverse to the above map, we send $u : C(b)$ to $\lambda x. (f_x)_* u$. In one direction we have

$$(\lambda x. f_x_* u)(b) \equiv (f_b)_* u = (\text{id}_b)_* u = u.$$

In the other direction, for any $x : B$ we have $f_x : \text{hom}_B(b, x)$, and thus for any $\phi : \prod_{x:B} C(x)$ we have $\lambda t. \phi(f_x(t)) : \text{hom}_{C(f_x)}(\phi(b), \phi(x))$. Since C is covariant, $\sum_{v:C(x)} \text{hom}_{C(f_x)}(\phi(b), v)$ is contractible, so

$$(\phi(x), \lambda t. \phi(f_x(t))) = ((f_x)_* \phi(b), \text{trans}_{f_x, u}),$$

and in particular $\phi(x) = (f_x)_* \phi(b)$. □

Lemma 9.8. *For any Segal type A and $a : A$, the type $\sum_{x:A} \text{hom}_A(a, x)$ has an initial object (a, id_a) .*

Proof. Let $x : A$ and $f : \text{hom}_A(a, x)$; we must show that

$$\text{hom}_{\sum_{x:A} \text{hom}_A(a, x)}((a, \text{id}_a), (x, f))$$

is contractible. By Theorem 4.3, this type is equivalent to

$$\langle \prod_{t:2} \text{hom}_A(a, f(t)) \Big|_{\text{id}_a}^0 \rangle$$

which is contractible by Proposition 8.4 since $\text{hom}_A(a, -)$ is covariant. □

Proof of Theorem 9.5. By Lemma 9.8, $\sum_{x:A} \text{hom}_A(a, x)$ has an initial object (a, id_a) . Thus Theorem 9.7 specializes to the desired result. □

A formula for the inverse of the dependent evid_a^C

$$\text{yon}_a^C : C(a, \text{id}_a) \rightarrow \left(\prod_{x:A} \prod_{f:\text{hom}_A(a, x)} C(x, f) \right)$$

can be extracted from the above proofs. Under the equivalent description of

$$\text{hom}_{\sum_{x:A} \text{hom}_A(a, x)}((a, \text{id}_a), (x, f))$$

in Lemma 9.8, a specific inhabitant of it is given by the connection square Λ_f from Proposition 3.5. Thus, we can write

$$\text{yon}_a^C(u, x, f) := (\Lambda_f)_* u. \tag{9.9}$$

We say that a covariant type family $C : A \rightarrow \mathcal{U}$ over a Segal type A is **representable** if there exists some $a : A$ and a family of equivalences over A :

$$\prod_{x:A} (\text{hom}_A(a, x) \simeq C(x)).$$

Proposition 9.10. *A covariant type family $C : A \rightarrow \mathcal{U}$ over a Segal type A is representable if and only if the type*

$$\sum_{x \in A} C(x)$$

has an initial object (a, u) , in which case

$$\text{yon}_a^C(u) : \prod_{x:A} (\text{hom}_A(a, x) \rightarrow C(x))$$

defines an equivalence.

Proof. If C is representable, then there is an equivalence

$$\phi : \prod_{x:A} (\text{hom}_A(a, x) \simeq C(x))$$

corresponding under the Yoneda lemma to a term $\text{evid}_a^C(\phi) : C(a)$ defined by $\text{evid}_a^C(\phi) = \phi(a, \text{id}_a)$. By Lemma 9.8, (a, id_a) is initial in $\sum_{x:A} \text{hom}_A(a, x)$. Transporting along the equivalence, we conclude that $(a, \phi(a, \text{id}_a)) : \sum_{x:A} C(x)$ is initial and moreover that

$$\text{yon}_a^C(\phi(a, \text{id}_a)) = \text{yon}_a^C(\text{evid}_a^C(\phi)) = \phi$$

defines the postulated equivalence ϕ .

Conversely, if $(a, u) : \sum_{x:A} C(x)$ is initial then we argue that

$$\text{yon}_a^C(u) : \prod_{x:A} (\text{hom}_A(a, x) \rightarrow C(x))$$

defines an equivalence by showing that its fibers are contractible. From the definition of $\text{yon}_a^C(u)$ the fiber of

$$\text{yon}_a^C(u)_b : \text{hom}_A(a, b) \rightarrow C(b)$$

over $v : C(b)$ is

$$\sum_{f:\text{hom}_A(a,b)} f_*(u) =_{C(b)} v.$$

By Lemma 8.15 and Theorem 4.3

$$\begin{aligned} \left(\sum_{f:\text{hom}_A(a,b)} f_*(u) =_{C(b)} v \right) &\simeq \left(\sum_{f:\text{hom}_A(a,b)} \text{hom}_{C(f)}(u, v) \right) \\ &\simeq \text{hom}_{\sum_{x:A} C(x)}((a, u), (b, v)), \end{aligned}$$

which is contractible since (a, u) is initial. □

Yoneda lemmas for bisimplicial sets have recently been studied by [25, 13, 21], with similar conclusions. For instance, Proposition 9.10 above corresponds to [21, Theorem 5.6].

10. Rezk types

A Segal type is a type A whose hom-types $\text{hom}_A : A \rightarrow A \rightarrow \mathcal{U}$ are enhanced by a homotopically unique composition operation. A Rezk type is a Segal type that is “complete” or “univalent” in the sense that the identity type $x =_A y$ between any two terms is equivalent to the type of isomorphisms $x \cong_A y$ that we now introduce.

10.1 Isomorphisms Let A be a Segal type and consider $f : \text{hom}_A(x, y)$. As in ordinary category theory, we say f is an isomorphism if it has a two-sided composition inverse. However, as for functions in ordinary homotopy type theory [29, Chapter 4], more care is required to define a type of witnesses for the invertibility of f in such a way that it is contractible if it is inhabited. Guided by that experience, we define:

$$\text{iso}(f) := \left(\sum_{g:\text{hom}_A(y,x)} g \circ f = \text{id}_x \right) \times \left(\sum_{h:\text{hom}_A(y,x)} f \circ h = \text{id}_y \right)$$

and say that f is an isomorphism if this type is inhabited.

Proposition 10.1. *Let A be a Segal type and $f : \text{hom}_A(x, y)$. Then f is an isomorphism if and only if we have $g : \text{hom}_A(y, x)$ with $g \circ f = \text{id}_x$ and $f \circ g = \text{id}_y$.*

Proof. “Only if” is easy; take $h := g$. Conversely, from an inhabitant of $\text{iso}(f)$ we get g and h with $g \circ f = \text{id}_x$ and $f \circ h = \text{id}_y$, and then we can show

$$g = g \circ \text{id}_y = g \circ (f \circ h) = (g \circ f) \circ h = \text{id}_x \circ h = h$$

and therefore $f \circ g = \text{id}_y$ as well. □

Proposition 10.2. *Let A be a Segal type and $f : \text{hom}_A(x, y)$. Then the type $\text{iso}(f)$ is a proposition.*

Proof. If f is an isomorphism witnessed by a left inverse g and right inverse h , then for any $k : \text{hom}_A(z, x)$ we have $k = (g \circ f) \circ k = g \circ (f \circ k)$, and for any $\ell : \text{hom}_A(z, y)$ we have $\ell = (f \circ h) \circ \ell = f \circ (h \circ \ell)$. Therefore, the function $(f \circ -) : \text{hom}_A(z, x) \rightarrow \text{hom}_A(z, y)$ has both a left and a right inverse, and hence it is an equivalence [29, 4.3.3].

Since $\sum_{h:\text{hom}_A(y,x)} f \circ h = \text{id}_y$ is a fiber of $(f \circ -)$, it is therefore contractible. Similarly, the function $(- \circ f) : \text{hom}_A(y, z) \rightarrow \text{hom}_A(x, z)$ is an equivalence, so its fiber $\sum_{g:\text{hom}_A(y,x)} g \circ f = \text{id}_x$ is contractible, and hence $\text{iso}(f)$ is also contractible. In other words, if $\text{iso}(f)$ is inhabited, then it is contractible. Therefore, it is a proposition. □

Thus it makes sense to define **the type of isomorphisms** from x to y to be

$$(x \cong_A y) := \sum_{f:\text{hom}_A(x,y)} \text{iso}(f).$$

Consider now a pair of functions $f, g : X \rightarrow A$ where X is a type or shape and A is a Segal type. For any natural transformation $\alpha : \text{hom}_{X \rightarrow A}(f, g)$, if α is an isomorphism in $X \rightarrow A$ then clearly its components $\alpha_x : \text{hom}_A(f(x), g(x))$ are isomorphisms in A . Conversely:

Proposition 10.3. *Let X be a type or shape, let A be a Segal type, and consider $\alpha : \text{hom}_{X \rightarrow A}(f, g)$. Then the map*

$$\text{iso}(\alpha) \rightarrow \prod_{x:X} \text{iso}(\alpha_x)$$

is an equivalence. That is, a natural transformation is an isomorphism if and only if it is a pointwise isomorphism.

Proof. Since both sides are propositions, it suffices to assume $\prod_{x:X} \text{is iso}(\alpha_x)$ and prove that α is an isomorphism. To define $\beta : \text{hom}_{X \rightarrow A}(g, f)$, we must assume $t : \mathbb{2}$ and then $x : X$, and define $\beta_x(t)$; but since α_x is an isomorphism it has an inverse, so we can take $\beta_x(t) = (\alpha_x)^{-1}(t)$, i.e. $\beta_x = (\alpha_x)^{-1}$. To show $\beta \circ \alpha = \text{id}_f$, by function extensionality it suffices to show that $(\beta \circ \alpha)_x = (\text{id}_f)_x$; but this follows by Proposition 6.5 since $\beta_x \circ \alpha_x = \text{id}_{f(x)}$. Similarly we have $\alpha \circ \beta = \text{id}_g$. \square

This gives “isomorphism extensionality”:

Corollary 10.4. *For X a type or shape, A a Segal type, and $f, g : X \rightarrow A$, we have*

$$(f \cong_{A^X} g) \simeq \prod_{x:X} (fx \cong_A gx).$$

Proof. We have

$$\begin{aligned} \prod_{x:X} (fx \cong_A gx) &\equiv \prod_{x:X} \sum_{\alpha_x : \text{hom}_A(f(x), g(x))} \text{is iso}(\alpha_x) \\ &\simeq \sum_{\alpha : \prod_{x:X} \text{hom}_A(f(x), g(x))} \prod_{x:X} \text{is iso}(\alpha_x) \\ &\simeq \sum_{\alpha : \text{hom}_{X \rightarrow A}(f, g)} \prod_{x:X} \text{is iso}(\alpha_x) \\ &\simeq \sum_{\alpha : \text{hom}_{X \rightarrow A}(f, g)} \text{is iso}(\alpha) \\ &\equiv (f \cong_{A^X} g). \square \end{aligned}$$

10.2 Rezk-completeness Of course, id_x is always an isomorphism. Thus, path induction allows us to define

$$\text{idtoiso} : \prod_{x, y : A} (x =_A y) \rightarrow (x \cong_A y) \tag{10.5}$$

by reducing to the case where $x \equiv y$ and our equality is $\text{refl}_x : x =_A x$, which we map to $\text{id}_x : (x \cong_A x)$.

Definition 10.6. A Segal type A is **Rezk-complete** if (10.5) is an equivalence, in which case we say that A is a **Rezk type**.

When working with Rezk types, it is useful to observe that idtoiso mediates between the type-theoretic operations on paths and the category-theoretic operations on arrows.

Lemma 10.7. *If A is Segal and $C : A \rightarrow \mathcal{U}$ is covariant, while $e : x =_A y$, then for any $u : C(x)$ we have*

$$\text{idtoiso}(e)_* u = \text{transport}^C(e, u)$$

(The left-hand side is covariant transport along an arrow, while the right-hand side is homotopy-type-theoretic transport along an equality.)

Proof. By path induction on e : when $e \equiv \text{refl}$, both sides are equal to u . \square

Lemma 10.8. *If A and B are Segal and $f : A \rightarrow B$, while $e : x =_A y$, then*

$$f_{\#}(\text{idtoiso}(e)) = \text{idtoiso}(\text{ap}_f(e))$$

Proof. By path induction on e : when $e \equiv \text{refl}$, both sides are equal to id_{f_x} . \square

Rezk types, like Segal types, are closed under function spaces.

Proposition 10.9. *If A is a Rezk type so is $X \rightarrow A$ for any type or shape X .*

Proof. For $f, g : X \rightarrow A$, the map $\text{idtoiso}_{f,g}$ factors through (perhaps relative) function extensionality and isomorphism extensionality (Corollary 10.4) and the maps $\text{idtoiso}_{fx,gx}$ for A :

$$(f =_{A^X} g) \xrightarrow{\simeq} \left(\prod_{x:X} (fx =_A gx) \right) \xrightarrow{\prod_{x:X} \text{idtoiso}_{fx,gx}} \left(\prod_{x:X} fx \cong_A gx \right) \xrightarrow{\simeq} (f \cong_{A^X} g).$$

If A is a Rezk type, the middle map is an equivalence, hence so is the composite. \square

We now observe that if Rezk types are the “categories”, then discrete types are the “groupoids”.

Proposition 10.10. *A type is discrete if and only if it is Rezk and all its arrows are isomorphisms.*

Proof. Note that the composite

$$(x =_A y) \xrightarrow{\text{idtoiso}} (x \cong_A y) \longrightarrow \text{hom}_A(x, y)$$

is the map idtoarr from Definition 7.1. Since being an isomorphism is a proposition, the inclusion $(x \cong_A y) \rightarrow \text{hom}_A(x, y)$ is an embedding, and hence is an equivalence if and only if it is surjective (i.e. all arrows are isomorphisms). This gives “if”, since equivalences compose. On the other hand, if A is discrete, then the composite is in particular surjective, hence so is the second factor. Thus this second factor is an equivalence, and hence so is the first factor; this gives “only if”. \square

10.3 Representable isomorphisms As a corollary of the Yoneda lemma, we can prove:

Proposition 10.11. *If given a pair of terms $a, a' : A$ in a Segal type and a fiberwise equivalence*

$$\phi : \prod_{x:A} \text{hom}_A(a, x) \simeq \text{hom}_A(a', x)$$

then the corresponding term $\text{evid}_a^{\text{hom}_A(a', -)}(\phi) : \text{hom}_A(a', a)$ is an isomorphism. If A is a Rezk type, then $a' =_A a$.

Proof. Let $u := \text{evid}_a^{\text{hom}_A(a', -)}(\phi)$. Then as observed in Remark 9.4, we have

$$\phi = \text{yon}_a^{\text{hom}_A(a', -)}(u) = \lambda x. \lambda f. f \circ u.$$

By the same argument, for the fiberwise inverse equivalence $\phi_x^{-1} : \text{hom}_A(a', x) \rightarrow \text{hom}_A(a, x)$ we have

$$\phi^{-1} = \text{yon}_a^{\text{hom}_A(a', -)}(v) = \lambda x. \lambda f. f \circ v.$$

where $v := \text{evid}_a^{\text{hom}_A(a', -)}(\phi^{-1})$. Since the composite of these fiberwise equivalences is equal to the identity function, we then have in particular that $\text{id}_a = (\text{id}_a \circ u) \circ v = \text{id}_a \circ (u \circ v) = u \circ v$ by the associativity and identity laws; similarly, $\text{id}_{a'} = v \circ u$. Thus u is an isomorphism and if A is a Rezk type then u proves that $a' =_A a$. \square

11. Adjunctions

In this section we introduce several types of adjunction data between a pair of types A and B and then investigate comparisons between these adjunction notions in the case where A and B are Segal or Rezk types. This extends the similar inquiry concerning data defining an equivalence between types in [29, Chapter 4].

11.1 Notions of adjunction In ordinary category theory, there are two ways of defining an adjunction: by a natural isomorphism of hom-sets, or in terms of a unit and counit satisfying the triangle identities. For clarity, in this section we will refer to the first style as a *transposing adjunction* and the latter as a *diagrammatic adjunction*. Transposing adjunctions generalize to our synthetic context fairly easily.

Definition 11.1. A **transposing adjunction** between types A, B consists of functors $f : A \rightarrow B$ and $u : B \rightarrow A$ and a family of equivalences

$$\prod_{\substack{a:A \\ b:B}} \text{hom}_B(fa, b) \simeq \text{hom}_A(a, ub).$$

Similarly, a **transposing left adjoint** of a functor $u : B \rightarrow A$ consists of a functor $f : A \rightarrow B$ together with such a family of equivalences. A **transposing right adjoint** of a functor $f : A \rightarrow B$ is defined dually.

On the other hand, in any sort of higher category theory, the triangle identities for a diagrammatic adjunction become data that can be asked to satisfy higher coherence laws as in [24]. We will indicate the absence of such coherence with the prefix “quasi-”, intentionally recalling the use of “quasi-inverse” in [29] for an incoherent homotopy inverse.

Definition 11.2. A **quasi-diagrammatic adjunction** between types A, B consists of:

- a functor $u : B \rightarrow A$,
- a functor $f : A \rightarrow B$,
- a natural transformation $\eta : \text{hom}_{A \rightarrow A}(\text{Id}_A, uf)$, and
- a natural transformation $\epsilon : \text{hom}_{B \rightarrow B}(fu, \text{Id}_B)$ together with
- a witness $\alpha : \text{hom}_{B \rightarrow A}^2 \left(\begin{array}{ccc} \eta u & ufu & u\epsilon \\ u & \text{id}_u & u \end{array} \right)$ and
- a witness $\beta : \text{hom}_{A \rightarrow B}^2 \left(\begin{array}{ccc} f\eta & fuf & \epsilon f \\ f & \text{id}_f & f \end{array} \right)$.

Similarly, a **quasi-diagrammatic left adjoint** of a functor $u : B \rightarrow A$ consists of the last five data above, and dually.

Note that if A is Segal, then by Proposition 5.12, the last two data may be presented equally as homotopies $\alpha : u\epsilon \circ \eta u = \text{id}_u$ and $\beta : \epsilon f \circ f\eta = \text{id}_f$. We will frequently pass back and forth between these two points of view in §§11.2 and 11.3. We have phrased the definition using higher simplices because it makes the connection to the theory of [24] clearer; the incoherence of a quasi-diagrammatic adjunction, for instance, corresponds to the fact that Example 4.2.4 of [24] is not a “parental subcomputad”.

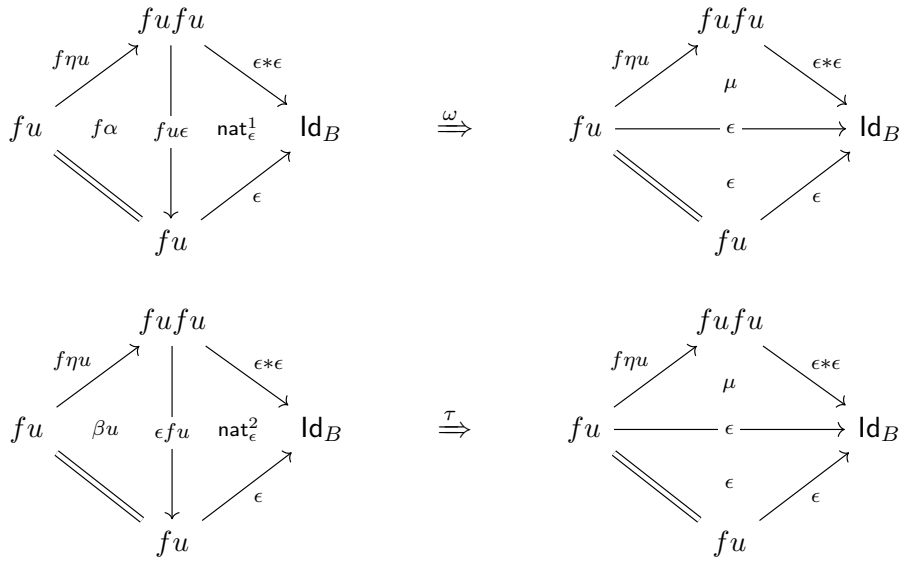


Figure 5: The 3-simplices in a half-adjoint diagrammatic adjunction

One of the main results of [24] is that while a fully coherent adjunction requires infinitely much data, that data is determined up to a contractible space of choices by various finite subcollections; these are the *parental subcomputads* of the generic adjunction $\underline{\text{Adj}}$. The formal framework of [24] uses simplicially enriched categories to represent $(\infty, 2)$ -categories, with their hom-spaces regarded as presenting quasi-categories. This can quite easily be translated into our setting, because our function-types, being types, have simplicial structure, and composition of functions in type theory is even strictly associative and unital.

As described in Examples 4.2.3 and 4.2.5 of [24], the four simplest parental subcomputads correspond in our framework to the following data:

1. A functor $f : A \rightarrow B$ only.
2. Functors $f : A \rightarrow B$ and $u : B \rightarrow A$ and a transformation $\epsilon : \text{hom}_{B \rightarrow B}(fu, \text{Id}_B)$.
3. Functors f and u , transformations η and ϵ , and the 2-simplex β .
4. Functors f and u , transformations η and ϵ , 2-simplices α and β , and 3-simplices ω and τ with one new common 2-simplex face μ .

Of these, the last is the first one that includes at least the data of a quasi-diagrammatic adjunction, so that it suffices to determine some kind of adjunction without further hypotheses. We name the corresponding structure in our setting by analogy to the “half-adjoint equivalences” of [29, §4.2]. The simplices μ, ω, τ are (like α and β) named as in [24, §§1.1 and 4.2].

Definition 11.3. A **half-adjoint diagrammatic adjunction** between types consists of a quasi-diagrammatic adjunction together with:

- A 2-simplex $\mu : \text{hom}_{B \rightarrow B}^2 \left(\begin{array}{ccc} f\eta u & fu fu & \epsilon * \epsilon \\ fu & \epsilon & \text{Id}_A \end{array} \right)$, where $\epsilon * \epsilon$ is the horizontal composite from §6.3.
- Two 3-simplices ω and τ with the boundaries shown in Figure 5, where the 2-simplex denoted ϵ is degenerate, and the 2-simplices nat_ϵ^1 and nat_ϵ^2 are the two halves of the Gray interchanger $\epsilon_{\#}(\epsilon)$ from §6.3.

If A is Segal, then by Corollary 5.22 the 3-simplices ω and τ can equivalently be regarded as equalities relating the following two concatenated equalities to the homotopy $(\epsilon * \epsilon) \circ f\eta u = \epsilon$ corresponding to μ :

$$(\epsilon * \epsilon) \circ f\eta u = (\epsilon \circ fu\epsilon) \circ f\eta u = \epsilon \circ (fu\epsilon \circ f\eta u) \stackrel{f\alpha}{=} \epsilon \circ \text{id}_{fu} = \epsilon \quad (11.4)$$

$$(\epsilon * \epsilon) \circ f\eta u = (\epsilon \circ \epsilon fu) \circ f\eta u = \epsilon \circ (\epsilon fu \circ f\eta u) \stackrel{\beta u}{=} \epsilon \circ \text{id}_{fu} = \epsilon. \quad (11.5)$$

That is, in a Segal type, the type of half-adjoint diagrammatic adjunctions extending a given quasi-diagrammatic adjunction is equivalent to

$$\sum_{\mu: (\epsilon * \epsilon) \circ f\eta u = \epsilon} ((11.4) = \mu) \times ((11.5) = \mu).$$

By contracting a based path space, this is equivalent to simply $((11.4) = (11.5))$, i.e. the type of 2-homotopies filling the following diagram:

$$\begin{array}{ccccc} & & (\epsilon \circ fu\epsilon) \circ f\eta u & \xrightarrow{\quad} & \epsilon \circ (fu\epsilon \circ f\eta u) & \xrightarrow{f\alpha} & \epsilon \circ \text{id}_{fu} & & \\ & \swarrow & & & & & & \searrow & \\ (\epsilon * \epsilon) \circ f\eta u & & & & & & & & \epsilon \\ & \searrow & & & & & & \swarrow & \\ & & (\epsilon \circ \epsilon fu) \circ f\eta u & \xrightarrow{\quad} & \epsilon \circ (\epsilon fu \circ f\eta u) & \xrightarrow{\beta u} & \epsilon \circ \text{id}_{fu} & & \end{array}$$

Since the concatenation $\epsilon \circ fu\epsilon = \epsilon * \epsilon = \epsilon \circ \epsilon fu$ consists of the naturality squares for ϵ at the components of ϵ , if we disregard the associativity and unit coherences we can write this as

$$\begin{array}{ccc} \epsilon \circ fu\epsilon \circ f\eta u & \xrightarrow{\text{nat}_\epsilon} & \epsilon \circ \epsilon fu \circ f\eta u. \\ & \searrow f\alpha & \swarrow \beta u \\ & \epsilon & \end{array}$$

We will see in §11.2 that half-adjoint diagrammatic adjunctions correspond to half-adjoint equivalences in the sense of [29, §4.2], justifying the name. On the other hand, the notion of “bi-invertible map” from [29, §4.3] suggests the following modification instead:

Definition 11.6. A **bi-diagrammatic adjunction** between types A, B consists of:

- a functor $u : B \rightarrow A$,
- a functor $f : A \rightarrow B$,
- a natural transformation $\eta : \text{hom}_{A \rightarrow A}(\text{Id}_A, uf)$, and
- two natural transformations $\epsilon, \epsilon' : \text{hom}_{B \rightarrow B}(fu, \text{Id}_B)$ together with
- a witness $\alpha : \text{hom}_{B \rightarrow A}^2 \left(\begin{array}{ccc} \eta u & ufu & u\epsilon \\ u & \text{id}_u & u \end{array} \right)$ and
- a witness $\beta : \text{hom}_{A \rightarrow B}^2 \left(\begin{array}{ccc} f\eta & fu f & \epsilon' f \\ f & \text{id}_f & f \end{array} \right)$.

Note that ϵ appears in α but ϵ' appears in β .

Our goal in the rest of this section is to compare all these kinds of adjunction, for Segal and Rezk types.

11.2 Adjunctions between Segal types We begin by observing that quasi-diagrammatic adjunctions suffice to induce transposing adjunctions. More precisely, we show that a quasi-diagrammatic adjunction corresponds exactly to the following.

Definition 11.7. A **quasi-transposing adjunction** between types A, B consists of functors $f : A \rightarrow B$ and $u : B \rightarrow A$ and a family of maps

$$\phi : \prod_{\substack{a:A \\ b:B}} \text{hom}_B(fa, b) \rightarrow \text{hom}_A(a, ub)$$

equipped with quasi-inverses, i.e. a family of maps

$$\psi : \prod_{\substack{a:A \\ b:B}} \text{hom}_A(a, ub) \rightarrow \text{hom}_B(fa, b)$$

and homotopies $\xi : \prod_{a,b,k} \phi_{a,b}(\psi_{a,b}(k)) = k$ and $\zeta : \prod_{a,b,\ell} \psi_{a,b}(\phi_{a,b}(\ell)) = \ell$.

Theorem 11.8. *Given Segal types A, B and functors $f : A \rightarrow B$ and $u : B \rightarrow A$, the type of quasi-transposing adjunctions between f and u is equivalent to the type of quasi-diagrammatic adjunctions between f and u .*

Proof. Each of the two types is a dependent sum type with four components. We will show that each of the four components is equivalent to a corresponding one on the other side.

By Proposition 8.13 and Remark 8.3, for any $a : A$ the type family $\text{hom}_A(a, u-) : B \rightarrow \mathcal{U}$ is covariant. Thus, by the Yoneda lemma, the map

$$\text{yon}_{fa}^{\text{hom}_A(a, u-)} : \text{hom}_A(a, ufa) \rightarrow \prod_{b:B} (\text{hom}_B(fa, b) \rightarrow \text{hom}_A(a, ub))$$

is an equivalence, and hence so is the induced map

$$\text{hom}_{A \rightarrow A}(\text{ld}_A, uf) \simeq \left(\prod_{a:A} \text{hom}_A(a, ufa) \right) \rightarrow \prod_{\substack{a:A \\ b:B}} (\text{hom}_B(fa, b) \rightarrow \text{hom}_A(a, ub)) \quad (11.9)$$

which sends η to ϕ^η defined by $\phi_{a,b}^\eta(k) := uk \circ \eta_a$. Similarly, the map

$$\text{hom}_{B \rightarrow B}(fu, \text{ld}_B) \simeq \left(\prod_{b:B} \text{hom}_B(fub, b) \right) \rightarrow \prod_{\substack{a:A \\ b:B}} (\text{hom}_A(a, ub) \rightarrow \text{hom}_B(fa, b)),$$

sending ϵ to ψ^ϵ defined by $\psi_{a,b}^\epsilon(\ell) := \epsilon_b \circ f\ell$, is an equivalence.

It remains, therefore, to show that if we fix η and ϵ , then we have equivalences

$$\text{hom}_{B \rightarrow A}^2 \left(\begin{array}{ccc} \eta u & & u\epsilon \\ u & \xrightarrow{ufu} & u \\ & \text{id}_u & \end{array} \right) \simeq \prod_{a,b,k} (\psi_{a,b}^\epsilon(\phi_{a,b}^\eta(k)) = k)$$

and dually. First note that since A is Segal,

$$\begin{aligned} \text{hom}_{B \rightarrow A}^2 \left(\begin{array}{ccc} \eta u & & u\epsilon \\ u & \xrightarrow{ufu} & u \\ & \text{id}_u & \end{array} \right) &\simeq (u\epsilon \circ \eta u = \text{id}_u) \\ &\simeq \prod_{b:B} (u\epsilon_b \circ \eta_{ub} = \text{id}_{ub}). \end{aligned}$$

Thus, it suffices to construct an equivalence

$$(u\epsilon_b \circ \eta_{ub} = \text{id}_{ub}) \simeq \prod_{a:A} \prod_{k:\text{hom}_A(a,ub)} (\phi_{a,b}^\epsilon(\psi_{a,b}^\eta(k)) = k) \quad (11.10)$$

for any $b : B$.

Now by Theorem 8.26 and Remark 8.3, the family $\lambda a. \lambda k. (\phi_{a,b}^\epsilon(\psi_{a,b}^\eta(k)) = k)$ is contravariant. Thus, by the contravariant form of the dependent Yoneda lemma (Theorem 9.5), we have an equivalence

$$\left(\phi_{ub,b}^\epsilon(\psi_{ub,b}^\eta(\text{id}_{ub})) = \text{id}_{ub} \right) \simeq \prod_{a:A} \prod_{k:\text{hom}_A(a,ub)} (\psi_{a,b}^\epsilon(\phi_{a,b}^\eta(k)) = k). \quad (11.11)$$

Moreover, we have

$$\begin{aligned} \phi_{ub,b}^\epsilon(\psi_{ub,b}^\eta(\text{id}_{ub})) &\equiv \phi_{ub,b}^\epsilon(\epsilon_b \circ f \text{id}_{ub}) \\ &\equiv u(\epsilon_b \circ f \text{id}_{ub}) \circ \eta_{ub} \\ &\equiv u(\epsilon_b \circ \text{id}_{fub}) \circ \eta_{ub} \\ &= u\epsilon_b \circ \eta_{ub}. \end{aligned}$$

Concatenating with this identity yields an equivalence

$$\left(u\epsilon_b \circ \eta_{ub} = \text{id}_{ub} \right) \simeq \left(\phi_{ub,b}^\epsilon(\psi_{ub,b}^\eta(\text{id}_{ub})) = \text{id}_{ub} \right)$$

which in combination with (11.11) gives (11.10).

The second equivalence, involving β and ζ , is defined similarly, by combining a dependent Yoneda equivalence

$$\left(\psi_{a,fa}^\epsilon(\phi_{a,fa}^\eta(\text{id}_{fa})) = \text{id}_{fa} \right) \simeq \prod_{b:B} \prod_{\ell:\text{hom}_B(fa,b)} (\phi_{a,b}^\epsilon(\psi_{a,b}^\eta(\ell)) = \ell).$$

and concatenation with the equality

$$\begin{aligned} \psi_{a,fa}^\epsilon(\phi_{a,fa}^\eta(\text{id}_{fa})) &\equiv \epsilon_{fa} \circ f(\text{id}_{fa} \circ \eta_a) \\ &= \epsilon_{fa} \circ f\eta_a. \square \end{aligned}$$

Corollary 11.12. *Any quasi-diagrammatic adjunction between Segal types induces a transposing adjunction.*

Proof. This follows directly from Theorem 11.8 and the fact that any quasi-inverse can be improved to a coherent equivalence, [29, Theorem 4.2.3]. \square

Our next goal is to show that the two improved versions of diagrammatic adjunctions really are suitably “coherent”. One of them is very straightforward.

Theorem 11.13. *Given Segal types A, B and functors $f : A \rightarrow B$ and $u : B \rightarrow A$, the type of bi-diagrammatic adjunctions between them is equivalent to the type of transposing adjunctions.*

Proof. In the definition of transposing adjunction, as is usual in homotopy type theory, we did not specify exactly which coherent notion of “equivalence” was meant, since given function extensionality all of them are equivalent. For the purposes of this theorem, we take it to mean

bi-invertible maps. We can then unwind the definition of transposing adjunction to consist of a family of maps

$$\phi : \prod_{\substack{a:A \\ b:B}} \text{hom}_B(fa, b) \rightarrow \text{hom}_A(a, ub)$$

and *two* families of maps

$$\psi, \psi' : \prod_{\substack{a:A \\ b:B}} \text{hom}_A(a, ub) \rightarrow \text{hom}_B(fa, b)$$

together with homotopies $\prod_{a,b,k} \phi_{a,b}(\psi_{a,b}(k)) = k$ and $\prod_{a,b,\ell} \psi'_{a,b}(\phi_{a,b}(\ell)) = \ell$. The same arguments as in Theorem 11.8 then identify these data with those of a bi-diagrammatic adjunction. \square

The other requires a bit more work.

Theorem 11.14. *Given Segal types A, B and functors $f : A \rightarrow B$ and $u : B \rightarrow A$, the type of half-adjoint diagrammatic adjunctions between them is equivalent to the type of transposing adjunctions.*

Proof. Just as in Theorem 11.13 we formulated transposing adjunctions using bi-invertible maps, here we formulate them using half-adjoint equivalences. Thus, it remains to show that the type of coherence data μ in a half-adjoint diagrammatic adjunction is equivalent, over the equivalences constructed in Theorem 11.8, to the type of families of half-adjoint coherence data for a quasi-transposing adjunction (ϕ, ψ, ξ, ζ) :

$$\prod_{b:B} \prod_{a:A} \prod_{\ell:\text{hom}_A(a,ub)} \left(\zeta_{a,b}^\beta(\psi_{a,b}^\epsilon(\ell)) = \text{ap}_{\psi_{a,b}^\epsilon}(\xi_{a,b}^\alpha(\ell)) \right) \quad (11.15)$$

Applying Theorem 8.26 and Remark 8.3 twice, we see that

$$\lambda a. \lambda \ell. \left(\zeta_{a,b}^\beta(\psi_{a,b}^\epsilon(\ell)) = \text{ap}_{\psi_{a,b}^\epsilon}(\xi_{a,b}^\alpha(\ell)) \right)$$

is contravariant. Thus, by the dependent Yoneda lemma, (11.15) is equivalent to

$$\prod_{b:B} \left(\zeta_{ub,b}^\beta(\psi_{ub,b}^\epsilon(\text{id}_{ub})) = \text{ap}_{\psi_{ub,b}^\epsilon}(\xi_{ub,b}^\alpha(\text{id}_{ub})) \right). \quad (11.16)$$

Now we need to analyze ξ and ζ more carefully in terms of α and β . By definition, $\lambda a. \xi_{a,b}^\alpha$ is the image of the concatenated equality

$$\phi_{ub,b}^\epsilon(\psi_{ub,b}^\eta(\text{id}_{ub})) \equiv u(\epsilon_b \circ f \text{id}_{ub}) \circ \eta_{ub} = u\epsilon_b \circ \eta_{ub} \stackrel{\alpha}{=} \text{id}_{ub} \quad (11.17)$$

under the inverse dependent Yoneda map

$$\text{yon}_{ub} : \left(\phi_{ub,b}^\epsilon(\psi_{ub,b}^\eta(\text{id}_{ub})) = \text{id}_{ub} \right) \rightarrow \prod_{a:A} \prod_{k:\text{hom}_A(a,ub)} (\psi_{a,b}^\epsilon(\phi_{a,b}^\eta(k)) = k).$$

which implies that $\text{evid}_{ub}(\lambda a. \xi_{a,b}^\alpha)$, i.e. $\xi_{ub,b}^\alpha(\text{id}_{ub})$, is equal to (11.17). Now $\psi_{a,b}^\epsilon(k) \equiv \epsilon_b \circ fk$, so $\text{ap}_{\psi_{ub,b}^\epsilon} = \text{ap}_{(\epsilon_b \circ -)} \circ \text{ap}_{f\#}$. Thus the right-hand side of (11.16) is equal to the concatenation

$$\begin{aligned} \psi_{ub,b}^\epsilon(\phi_{ub,b}^\epsilon(\psi_{ub,b}^\eta(\text{id}_{ub}))) &\equiv \epsilon_b \circ f(u(\epsilon_b \circ \text{id}_{fub}) \circ \eta_{ub}) \\ &= \epsilon_b \circ f(u\epsilon_b \circ \eta_{ub}) \\ &\stackrel{\epsilon_b \circ f\alpha}{=} \epsilon_b \circ f \text{id}_{ub} \\ &\equiv \epsilon_b \circ \text{id}_{fub} \end{aligned} \quad (11.18)$$

in which the two non-judgmental equalities are obtained by $\mathbf{ap}_{f\#}$ followed by $\mathbf{ap}_{(\epsilon_b \circ -)}$ from those in (11.17). There are other ways to define an equality meriting the name “ $\epsilon_b \circ f\alpha$ ”, but they can all easily be shown to be equal.

Similarly, $\zeta_{a,b}^\beta$ is the image of the concatenated equality

$$\psi_{a,fa}^\epsilon(\phi_{a,fa}^\eta(\mathbf{id}_{fa})) \equiv \epsilon_{fa} \circ f(\mathbf{id}_{fa} \circ \eta_a) = \epsilon_{fa} \circ f\eta_a \stackrel{\beta}{=} \mathbf{id}_{fa} \quad (11.19)$$

under the inverse dependent Yoneda map

$$\mathbf{yon}_{fa} : \left(\psi_{a,fa}^\epsilon(\phi_{a,fa}^\eta(\mathbf{id}_{fa})) = \mathbf{id}_{fa} \right) \rightarrow \prod_{b:B} \prod_{\ell: \mathbf{hom}_B(fa,b)} (\phi_{a,b}^\epsilon(\psi_{a,b}^\eta(\ell)) = \ell).$$

As we saw in §9, the latter can be defined by $\mathbf{yon}_{fa}(e, b, \ell) := (\Lambda_\ell)_*(e)$.

Now, for any $b : B$ and $\ell : \mathbf{hom}_B(fa, b)$, define

$$\begin{aligned} C(a, b, \ell) &:= \left(\phi_{a,b}^\epsilon(\psi_{a,b}^\eta(\ell)) = \ell \right) \\ &\equiv \left(\epsilon_b \circ f(u\ell \circ \eta_a) = \ell \right) \\ D(a, b, \ell) &:= \sum_{m: \mathbf{hom}_A(a, ub)} \mathbf{hom}_A^2 \left(\begin{array}{c} \eta_a \\ a \xleftarrow{m} ufa \xrightarrow{u\ell} ub \end{array} \right) \times \mathbf{hom}_B^2 \left(\begin{array}{c} fm \\ fa \xleftarrow{\ell} fub \xrightarrow{\epsilon_b} b \end{array} \right). \end{aligned}$$

Then we have $C(a, b, \ell) \simeq D(a, b, \ell)$, by contracting the first two components of $D(a, b, \ell)$, which are contractible since A is Segal, and then applying Proposition 5.12. Specifically, the map $D(a, b, \ell) \rightarrow C(a, b, \ell)$ takes (m, γ, δ) to the concatenated equality

$$\epsilon_b \circ f(u\ell \circ \eta_a) \stackrel{\epsilon_b \circ f\gamma}{=} \epsilon_b \circ fm \stackrel{\delta}{=} \ell.$$

(This can be proven easily by assuming that γ comes from an equality and doing path induction on it.) In particular, since C is covariant by Theorem 8.26 and Remark 8.3, so is D .

We are interested in two cases of this equivalence:

- For any $a : A$, we have $(\eta_a, s_0\eta_a, \beta_a) : D(a, fa, \mathbf{id}_{fa})$. The corresponding element of $C(a, fa, \mathbf{id}_{fa})$ is (11.19). Thus, the element of $D(a, b, \ell)$ corresponding to $\mathbf{yon}_{fa}(e, b, \ell) := (\Lambda_\ell)_*(e)$, when e is (11.19), is the covariant transport $(\Lambda_\ell)_*(\eta_a, s_0\eta_a, \beta_a)$ in the type family $D(a, -, -)$.
- For any $b : B$, we have $(\mathbf{id}_{ub}, \alpha_b, s_1\epsilon_b) : D(ub, b, \epsilon_b)$. Transporting this along the equality $p : \epsilon_b = \epsilon_b \circ \mathbf{id}_{fub} \equiv \psi_{ub,b}^\eta(\mathbf{id}_{ub})$ to obtain an element of $D(ub, b, \psi_{ub,b}^\eta(\mathbf{id}_{ub}))$, and passing back into $C(ub, b, \psi_{ub,b}^\eta(\mathbf{id}_{ub}))$, we obtain our computation (11.18) of $\mathbf{ap}_{\psi_{ub,b}^\eta}(\xi_{ub,b}^\alpha(\mathbf{id}_{ub}))$.

Our goal, therefore, is to identify the equality type

$$(\Lambda_{\epsilon_b \circ \mathbf{id}_{fub}})_*(\eta_{ub}, s_0\eta_{ub}, \beta_{ub}) = \mathbf{transport}(p, (\mathbf{id}_{ub}, \alpha_b, s_1\epsilon_b)).$$

or equivalently by Lemma 10.7 the simpler

$$(\Lambda_{\epsilon_b})_*(\eta_{ub}, s_0\eta_{ub}, \beta_{ub}) = (\mathbf{id}_{ub}, \alpha_b, s_1\epsilon_b).$$

Now by Lemma 8.15, this is equivalent to

$$\mathbf{hom}_{D(ub)(\epsilon_b, \Lambda_{\epsilon_b})}((\eta_{ub}, s_0\eta_{ub}, \beta_{ub}), (\mathbf{id}_{ub}, \alpha_b, s_1\epsilon_b)). \quad (11.20)$$

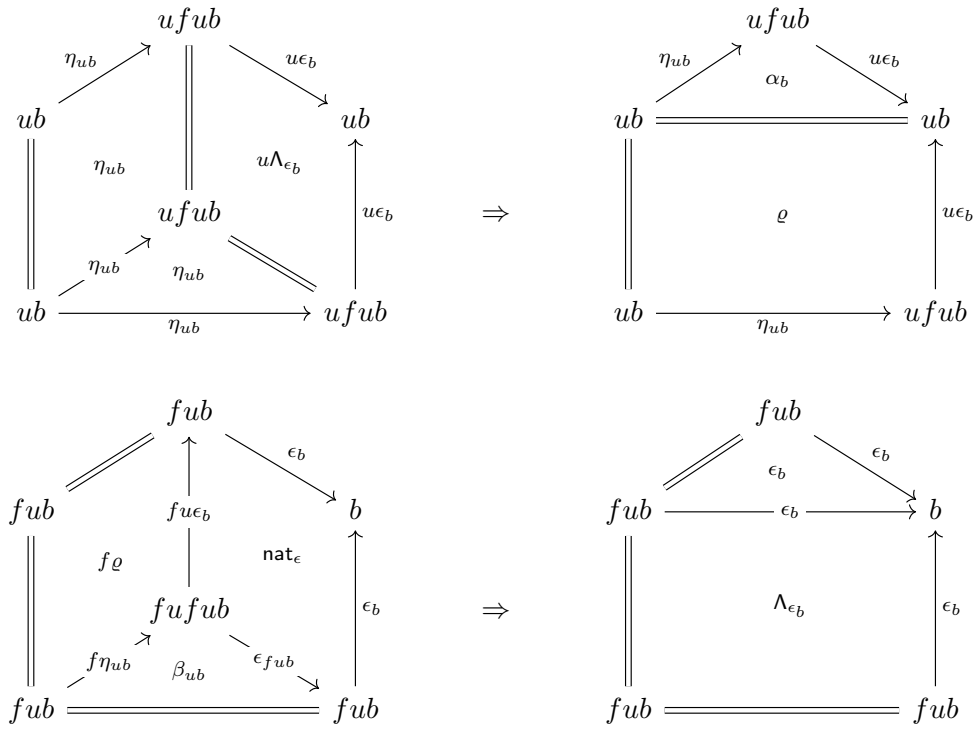


Figure 6: Two prisms

The notation $D(ub)(\epsilon_b, \Lambda_{\epsilon_b})$ means we consider dependent arrows in the type family $D(ub, -, -)$ over the arrow

$$(\epsilon_b, \Lambda_{\epsilon_b}) : \text{hom}_{\sum_{b':B} \text{hom}_B(fub, b')}((fub, \text{id}_{fub}), (b, \epsilon_b)).$$

Compiling this out, and using the equivalences of §4, we see that an element of (11.20) consists of two “triangular prisms” $\Delta^2 \times \Delta^1 \rightarrow A$ and $\Delta^2 \times \Delta^1 \rightarrow B$, with some elements of their boundary fixed as shown in Figure 6.

The square ϱ is not fixed, but must be the same in both prisms; and none of the interior simplices (not shown) are fixed. Squares or 2-simplices marked with the name of an arrow are constant/degenerate in the other direction, and the square denoted nat_ϵ is the naturality square for ϵ at itself, as constructed in §6.2.

Now, the boundary data of the top prism (in A) that is fixed consists of a “trough”

$$(\Lambda_1^2 \times \Delta^1) \cup_{(\Lambda_1^2 \times \partial\Delta^1)} (\Delta^2 \times \partial\Delta^1) \rightarrow A.$$

The inclusion of the trough into the prism $\Delta^2 \times \Delta^1$ is the pushout product of $\Lambda_1^2 \rightarrow \Delta^2$ and $\partial\Delta^1 \rightarrow \Delta^1$. Thus, by Proposition 5.20, the type of fillers (consisting of ϱ and all the inner simplices in the top prism) is contractible. Thus, it does not affect the homotopy type of (11.20), so in identifying the latter we are free to fix any particular such filler. We choose the following one:

$$\begin{aligned} \Delta^2 \times \Delta^1 &\rightarrow A \\ \{\langle t_1, t_1 \rangle : 2 \times 2 \mid t_2 \leq t_1\} \times \{t_3 : 2 \mid \top\} &\rightarrow A \\ \langle \langle t_1, t_2 \rangle, t_3 \rangle &\mapsto \begin{cases} \alpha_b(t_2, t_1) & t_2 \leq t_3 \\ \alpha_b(t_3, t_1) & t_3 \leq t_2 \end{cases} \end{aligned}$$

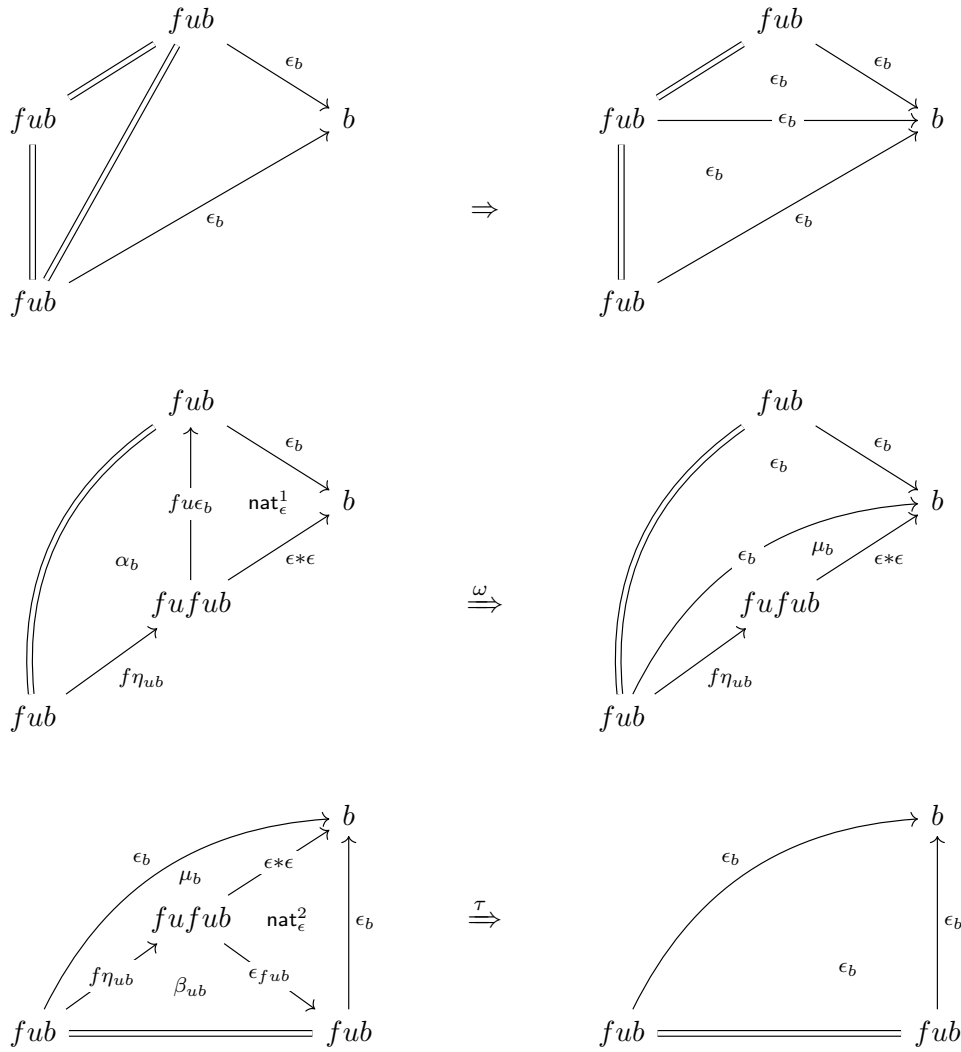


Figure 7: The three 3-simplices in a prism

It is straightforward to verify that this has the correct boundary. It determines ϱ to be the following square:

$$\begin{array}{ccc}
 ub & \xlongequal{\quad} & ub \\
 \parallel & \nearrow \alpha_b & \uparrow u\epsilon_b \\
 ub & \xrightarrow{\eta_{ub}} & ufub
 \end{array}$$

Now the second prism has its entire boundary fixed. As noted in §3.2, a prism consists of three 3-simplices glued along two common boundary 2-simplices. When the boundary of the prism is fixed, the “upper” of these 3-simplices has a 3-1-horn on its boundary fixed, the “lower” one has a 3-2-horn on its boundary fixed, while the “middle” one has only two faces of its boundary fixed. By Proposition 5.21, the types of 3-simplex fillers for 3-1-horns and 3-2-horns are contractible, so in determining the homotopy type of prisms we may assume a particular filler for the upper and/or lower horns.

In our case, the upper 3-1-horn has an obvious filler given by a doubly degenerate 3-simplex on ϵ_b . If we fill this in, the remaining two 3-simplices and their common boundary 2-simplex are

exactly the data of ω , τ , and μ from Definition 11.3 (evaluated at $b : B$). Figure 7 shows all three 3-simplices roughly as they sit inside the prism. Thus, the type of such prisms is equivalent to the type of half-adjoint diagrammatic adjunctions. \square

We can therefore conclude:

Corollary 11.21. *Given Segal types A, B and functors $f : A \rightarrow B$ and $u : B \rightarrow A$ along with a natural transformation $\eta : \text{hom}_{A \rightarrow A}(\text{Id}_A, uf)$, the following types are equivalent propositions.*

- (i) *The type of witnesses that $\lambda k. uk \circ \eta_a : \text{hom}_B(fa, b) \rightarrow \text{hom}_A(a, ub)$ is an equivalence for all a, b .*
- (ii) *The type of $(\epsilon, \epsilon', \alpha, \beta)$ extending (f, u, η) to a bi-diagrammatic adjunction.*
- (iii) *The type of $(\epsilon, \alpha, \beta, \mu, \omega, \tau)$ extending (f, u, η) to a half-adjoint diagrammatic adjunction.*
- (iv) *The propositional truncation of the type of $(\epsilon, \alpha, \beta)$ extending (f, u, η) to a quasi-diagrammatic adjunction.*

Proof. We have seen that when (i) is expressed using bi-invertibility it is equivalent to (ii), and that when it is expressed using half-adjoint equivalences it is equivalent to (iii). But (i) is always a proposition, however expressed. Finally, (iv) is a proposition by definition, which implies (ii) and is implied by (iii). \square

In other words, if a given transformation η is the unit of an adjunction, then that adjunction is uniquely determined up to a contractible space of choices. This corresponds to the dual of the fact mentioned in §11.1 that (f, u, ϵ) is a parental subcomputad of $\underline{\text{Adj}}$. Similarly, the fact that $(f, u, \eta, \epsilon, \beta)$ is a parental subcomputad corresponds to the dual of the following:

Corollary 11.22. *Given data $(f, u, \eta, \epsilon, \alpha)$ as in a quasi-diagrammatic adjunction, the following types are equivalent propositions:*

- (i) *The type of (ϵ', β) extending $(f, u, \eta, \epsilon, \alpha)$ to a bi-diagrammatic adjunction.*
- (ii) *The type of $(\beta, \mu, \omega, \tau)$ extending $(f, u, \eta, \epsilon, \alpha)$ to a half-adjoint diagrammatic adjunction.*
- (iii) *The propositional truncation of the type of β extending $(f, u, \eta, \epsilon, \alpha)$ to a quasi-diagrammatic adjunction.*

Proof. Since there is a map from (ii) to (i) (take $\epsilon' := \epsilon$) that becomes an equivalence when summed over ϵ and α , it is already an equivalence. Moreover, the proof of Theorem 11.13 actually shows that given (f, u, η) , the types of (ϵ, α) and (ϵ', β) are equivalent to the types $\text{linv}(\phi^\eta)$ and $\text{rinv}(\phi^\eta)$ of left and right inverses to ϕ^η respectively (see [29, Definition 4.2.7]). Since these types are both contractible as soon as they are both inhabited [29, Lemma 4.2.9], it follows that (i) is a proposition, hence so is (ii). Finally, (iii) is a proposition by definition, which implies (i) and is implied by (ii). \square

We would also like to know that if a given functor u has a left adjoint, then the entire adjunction is likewise uniquely determined, corresponding to the dual of the fact that (f) itself is already a parental subcomputad. However, since uniqueness of a functor $f : A \rightarrow B$ involves equalities in B , for this we need to assume our types are not just Segal but Rezk.

11.3 Adjunctions between Rezk types Under the additional hypothesis that the domain of a functor is Rezk and not just Segal, we can prove:

Theorem 11.23. *Given a Segal type A and a Rezk type B , and a functor $u : B \rightarrow A$, the following types are equivalent propositions.*

- (i) *The type of transposing left adjoints of u .*
- (ii) *The type of functors $f : A \rightarrow B$ and transformations $\eta : \text{hom}_{A \rightarrow A}(\text{Id}_A, uf)$ such that $\lambda k. uk \circ \eta_a : \text{hom}_B(fa, b) \rightarrow \text{hom}_A(a, ub)$ is an equivalence for all a, b .*
- (iii) *The type of half-adjoint diagrammatic left adjoints of u .*
- (iv) *The type of bi-diagrammatic left adjoints of u .*
- (v) *The propositional truncation of the type of quasi-diagrammatic left adjoints of u .*

Proof. The equivalence between (i) and (ii) follows by passing across the single Yoneda equivalence (11.9), while Corollary 11.21 implies that (ii), (iii), and (iv) are equivalent. And (v) is a proposition that is implied by (iii), and will imply (iv) as soon the latter is a proposition.

Thus, it suffices to show that (ii) is a proposition. But this is equivalent to

$$\sum_{f:A \rightarrow B} \sum_{\eta: \prod_{a:A} \text{hom}_A(a, ufa)} \prod_{a:A} \prod_{b:B} \text{isEquiv}(\lambda k. uk \circ \eta_a)$$

and this is equivalent to

$$\prod_{a:A} \sum_{f_a:B} \sum_{\eta_a: \text{hom}_A(a, ufa)} \prod_{b:B} \text{isEquiv}(\lambda k. uk \circ \eta_a).$$

Thus, since a product of propositions is a proposition, it suffices to prove that

$$\sum_{f_a:B} \sum_{\eta_a: \text{hom}_A(a, ufa)} \prod_{b:B} \text{isEquiv}(\lambda k. uk \circ \eta_a)$$

is a proposition for all $a : A$.

Note that this is the type of “universal arrows” from a to the functor u ; thus we are now reduced to essentially the usual proof of uniqueness of such universal arrows. Let (f_a, η_a, ω) and (f'_a, η'_a, ω') be two elements of this type. Since ω and ω' belong to propositions, we can ignore them for purposes of proving equality; what they give us is that the maps

$$\begin{aligned} \lambda k. uk \circ \eta_a &: \text{hom}_B(f_a, b) \rightarrow \text{hom}_A(a, ub) \\ \lambda k. uk \circ \eta'_a &: \text{hom}_B(f'_a, b) \rightarrow \text{hom}_A(a, ub) \end{aligned}$$

are equivalences for any $b : B$. Taking $b := f'_a$ in the first equivalence, and applying its inverse to η'_a , we obtain $m : \text{hom}_B(f_a, f'_a)$ such that $um \circ \eta_a = \eta'_a$. Then taking $b := f_a$ in the second equivalence, and applying its inverse to η_a , we obtain $n : \text{hom}_B(f'_a, f_a)$ such that $un \circ \eta'_a = \eta_a$. Thus, $u(m \circ n) \circ \eta'_a = \eta'_a$ and $u(n \circ m) \circ \eta_a = \eta_a$, so by the injectivity of equivalences, m and n are inverse isomorphisms in B .

Now since B is Rezk, we have $e : f_a = f'_a$ such that $\text{idtoiso}(e) = m$. By the characterization of equalities in Σ -types, it suffices to show that

$$\text{transport}^{\lambda b. \text{hom}_A(a, ub)}(e, \eta_a) = \eta'_a.$$

But using Lemmas 10.7 and 10.8, we have

$$\begin{aligned} \text{transport}^{\lambda b. \text{hom}_A(a, ub)}(e, \eta_a) &= \text{transport}^{\lambda x. \text{hom}_A(a, x)}(\text{ap}_u(e), \eta_a) \\ &= \text{idtoiso}(\text{ap}_u(e)) \circ \eta_a \\ &= u(\text{idtoiso}(e)) \circ \eta_a \\ &= um \circ \eta_a \\ &= \eta'_a. \square \end{aligned}$$

In other words, for Rezk types (regarded as synthetic $(\infty, 1)$ -categories), adjoints are literally unique, not just “unique up to isomorphism”. This should be compared with [29, Lemma 9.3.2], which proves an analogous fact for 1-categories *defined* internally to ordinary homotopy type theory (rather than axiomatized synthetically) and satisfying a similar Rezk-completeness condition.

Appendix A: Semantics of simplicial type theory

In this section we review the model of homotopy type theory in the category of Reedy fibrant bisimplicial sets from [27] and describe how this category also models the simplicial type theory of cubes, topes, and shapes. We will not give a complete proof, but only sketch the main ideas. We then prove that Segal types correspond exactly to the *Segal spaces* in this model, while Rezk types correspond to the complete Segal spaces [22], which are also called *Rezk spaces*.

A.1 Reedy fibrations of bisimplicial sets The category $\mathbf{sSet} := \mathbf{Set}^{\Delta^{\text{op}}}$ of simplicial sets embeds in two “orthogonal” ways into the category $\mathbf{ssSet} := \mathbf{Set}^{\Delta^{\text{op}} \times \Delta^{\text{op}}}$ of bisimplicial sets. Via the isomorphism $\mathbf{ssSet} \cong \mathbf{sSet}^{\Delta^{\text{op}}}$ that expresses a bisimplicial set X as a simplicial space, we regard $X_{m,n}$ as the set of n -simplices in the m th space of the simplicial object $X : \Delta^{\text{op}} \rightarrow \mathbf{sSet}$.

To define these two embeddings use the **external product bifunctor**

$$\mathbf{sSet} \times \mathbf{sSet} \xrightarrow{\square} \mathbf{ssSet} \quad (A \square B)_{m,n} := A_m \times B_n.$$

Note that $\Delta^m \square \Delta^n$ is the functor represented by the object $(m, n) \in \Delta \times \Delta$. In particular, using exponential notation for the internal hom in \mathbf{ssSet} , we have

$$(Y^X)_{m,n} = \mathbf{ssSet}(X \times (\Delta^m \square \Delta^n), Y).$$

Definition A.1 (the discrete and constant embeddings). Fixing one variable to be the point, we obtain embeddings

$$\text{disc} : \mathbf{sSet} \xrightarrow{-\square\Delta^0} \mathbf{ssSet} \quad \text{const} : \mathbf{sSet} \xrightarrow{\Delta^0\square-} \mathbf{ssSet}$$

of simplicial sets as **discrete** and **constant** bisimplicial sets, respectively. The discrete simplicial spaces have the form of functors $\Delta^{\text{op}} \rightarrow \mathbf{Set} \hookrightarrow \mathbf{sSet}$, while the constant simplicial spaces have the form of functors $\Delta^{\text{op}} \rightarrow \mathbf{1} \rightarrow \mathbf{sSet}$. The discrete embedding positions the data of a simplicial set in the “categorical” direction, while the constant embedding positions the data in the “spacial” direction.

The bifunctor $-\square-$ is biclosed. Under the identification $\mathbf{ssSet} \cong \mathbf{sSet}^{\Delta^{\text{op}}}$ described above, the left closure

$$A \square B \rightarrow X \quad \iff \quad B \rightarrow \{A, X\}$$

is the limit of X weighted by the simplicial set $A \in \mathbf{Set}^{\Delta^{\text{op}}}$; in particular, $\{\Delta^m, X\} \cong X_m$, the m th column of X .

Definition A.2. A bisimplicial set $X \rightarrow Y$ is a **Reedy fibration** if and only if for all $m \geq 0$ the induced map

$$\{\Delta^m, X\} \rightarrow \{\partial\Delta^m, X\} \times_{\{\partial\Delta^m, Y\}} \{\Delta^m, Y\}$$

on weighted limits is a Kan fibration in \mathbf{sSet} . A bisimplicial set X is **Reedy fibrant** just when the unique map $X \rightarrow 1$ is a Reedy fibration, which is the case when

$$\{\Delta^m, X\} \rightarrow \{\partial\Delta^m, X\}$$

is a Kan fibration.

Any bifunctor, such as \square , whose codomain has pushouts has an associated pushout product; in our case this defines a biclosed bifunctor

$$\mathbf{sSet}^2 \times \mathbf{sSet}^2 \xrightarrow{\widehat{\square}} \mathbf{ssSet}^2.$$

The set of maps

$$\{(\partial\Delta^m \hookrightarrow \Delta^m) \widehat{\square} (\partial\Delta^n \hookrightarrow \Delta^n)\}_{m,n \geq 0}$$

defines a set of generating Reedy cofibrations for \mathbf{ssSet} . A map of bisimplicial sets is a **Reedy trivial fibration** if and only if it has the right lifting property with respect to this set of maps.

Theorem A.3 (Shulman [27]). *The Reedy model structure on bisimplicial sets defined relative to the Quillen model structure on simplicial sets models intensional type theory with dependent sums, dependent products, identity types, and as many univalent universes as there are inaccessible cardinals greater than \aleph_0 .*

In the bisimplicial sets model, a dependent type family $C : A \rightarrow \mathcal{U}$ is modeled by a Reedy fibration $C \twoheadrightarrow A$, which we denote using an arrow “ \twoheadrightarrow ” for emphasis.

The Reedy fibrations enjoy the following important “Leibniz closure” property.

Lemma A.4. *If $i : U \rightarrow V$ is a monomorphism (equivalently, a cofibration) of bisimplicial sets and $p : X \twoheadrightarrow Y$ is a Reedy fibration then the map*

$$\langle X^i, p^V \rangle : X^V \rightarrow X^U \times_{Y^U} Y^V,$$

which we denote by $\widehat{\langle i, p \rangle}$, is a Reedy fibration, whose domain and codomain are Reedy fibrant if X and Y are, and which is a weak equivalence if p is.

Proof. By usual adjunction arguments, it suffices to prove that if $i : U \rightarrow V$ and $j : A \rightarrow B$ are cofibrations of bisimplicial sets, then the pushout product map $i \widehat{\times} j$ is a cofibration that is acyclic (i.e. a levelwise weak equivalence) if j is.

$$\begin{array}{ccc}
 U \times A & \xrightarrow{i \times 1_A} & V \times A \\
 \downarrow 1_U \times j & & \downarrow k \\
 U \times B & \longrightarrow & \bullet \\
 & \searrow i \widehat{\times} j & \nearrow 1_V \times j \\
 & & V \times B
 \end{array}$$

$\xrightarrow{i \times 1_B}$

All the solid arrows in this diagram are monomorphisms and the outer square is a pullback; thus so is the dashed arrow, being a “union of subobjects” of $V \times B$.

If j is acyclic, then since products of simplicial sets preserve weak equivalences, so do products of bisimplicial sets; hence $1_U \times j$ and $1_V \times j$ are weak equivalences. Thus the map $1_U \times j$ is an acyclic cofibration so its pushout, the map denoted k in the diagram, is again a weak equivalence. Thus, by the 2-out-of-3 property, $i \widehat{\times} j$ is a weak equivalence as well. \square

A.2 Modeling type theory with shapes The usual approach to modeling dependent type theory in a category \mathcal{C} is to exhibit a *comprehension category* over \mathcal{C} , which is a Grothendieck fibration $\mathcal{T} \rightarrow \mathcal{C}$ equipped with a functor over \mathcal{C} :

$$\begin{array}{ccc} \mathcal{T} & \xrightarrow{\quad} & \mathcal{C}^2 \\ & \searrow & \swarrow \text{cod} \\ & \mathcal{C} & \end{array}$$

that preserves cartesian arrows. In the homotopy-theoretic context, $\mathcal{T} \hookrightarrow \mathcal{C}^2$ is the subcategory of fibrations. The categorical structure of \mathcal{C} of interest induces similar structure on \mathcal{T} , to which one applies a *coherence theorem* such as [18] to obtain “strictly stable” structure on a split comprehension category equivalent to \mathcal{T} . Finally, one constructs a similar split comprehension category with strictly stable structure out of the syntax of type theory, taking the base category \mathcal{C} to be the contexts and the total category \mathcal{T} to be the types-in-context, and proves an “initiality theorem” that it is the initial such, and hence maps uniquely into the one constructed from the desired model \mathcal{C} .

Of these steps, the initiality theorem is commonly neglected; the proofs in known cases are universally expected to generalize to all other cases, but there is as yet no general theorem. Similarly, the coherence method of [18] is not yet a general theorem but has to be proven separately for each kind of type-theoretic structure. As our goal here is only to give a sketch of the semantics, we will omit both of these proofs; we confine ourselves to describing informally the relevant comprehension categories and explaining how both the syntax and the semantics yield examples.

To start with, since our type theory has three layers, our comprehension categories must also have three layers. The cube and tope layers have no “intra-layer dependencies”, so they do not require a full comprehension category structure individually; instead we can make do with a simple category with products. This does involve blurring the line between context extension and cartesian product of types (i.e. we identify $t : I, s : J$ with $\langle t, s \rangle : I \times J$), but it is common and unproblematic.¹² The dependency between levels is encoded with fibrations as in [7]. This leads to:

Definition A.5. A **comprehension category with shapes** is a tower of fibrations

$$\begin{array}{ccc} \mathcal{T} & \xrightarrow{\quad} & (\mathcal{C}_2)_{\mathcal{C}_1}^2 \\ & \searrow & \swarrow \text{cod} \\ & \mathcal{C}_2 & \\ & \downarrow \pi_2 & \\ & \mathcal{C}_1 & \\ & \downarrow \pi_1 & \\ & \mathcal{C}_0 & \end{array}$$

in which \mathcal{C}_0 has finite products, \mathcal{C}_1 has fiberwise finite products (i.e. its fibers have finite products preserved by reindexing), $(\mathcal{C}_2)_{\mathcal{C}_1}^2$ denotes the category of arrows in \mathcal{C}_2 that map to identities in \mathcal{C}_1 (and all commutative squares between them), and $\mathcal{T} \rightarrow (\mathcal{C}_2)_{\mathcal{C}_1}^2$ preserves cartesian arrows.

¹²Otherwise we could talk about “cartesian multicategories”.

Our type theory with shapes as described in §2 yields a comprehension category with shapes in which:

- The objects of \mathcal{C}_0 are the contexts of cubes, and the morphisms are tuples of terms modulo the equivalence relation of derivable equality in tope logic, i.e. if $\Xi \vdash (t \equiv s)$ then t and s represent the same morphism in \mathcal{C}_0 .
- The objects of \mathcal{C}_1 are contexts of topes-in-context, i.e. lists ϕ_1, \dots, ϕ_n where $\Xi \vdash \phi_i$ tope for each i , with reindexing by substitution along cube-morphisms. The morphisms in each fiber are entailments $\Xi \mid \Phi \vdash \psi$.
- The objects of \mathcal{C}_2 are contexts of types in context, i.e. the judgment that we wrote as $\Xi \mid \Phi \vdash \Gamma$ ctx, and its morphisms are tuples of terms in the type theory modulo judgmental equality.
- The objects of \mathcal{T} are types-in-context, i.e. judgments $\Xi \mid \Phi \mid \Gamma \vdash A$ type, and its morphisms are terms. The functor $\mathcal{T} \rightarrow (\mathcal{C}_2)_{\mathcal{C}_1}^2$ extends a context by a type.

On the other hand, the bisimplicial set model yields a comprehension category with shapes in which:

- \mathcal{C}_0 is the category of simplicial sets of the form $(\Delta^1)^n$, regarded as spatially-discrete bisimplicial sets.
- \mathcal{C}_1 is the category of monomorphisms of simplicial sets (regarded as spatially-discrete bisimplicial sets) whose codomain is of the form $(\Delta^1)^n$, with the projection $\mathcal{C}_1 \rightarrow \mathcal{C}_0$ the codomain functor.
- \mathcal{C}_2 is the category of diagrams $\Gamma \twoheadrightarrow \Phi \twoheadrightarrow I$ where $\Phi \twoheadrightarrow I$ is an object of \mathcal{C}_1 and $\Gamma \twoheadrightarrow \Phi$ is any Reedy fibration of bisimplicial sets.
- \mathcal{T} is the category of diagrams $A \twoheadrightarrow \Gamma \twoheadrightarrow \Phi \twoheadrightarrow I$, where $\Gamma \twoheadrightarrow \Phi \twoheadrightarrow I$ is as in \mathcal{C}_2 and $A \twoheadrightarrow \Gamma$ is a Reedy fibration.

The discrete embedding of sets in simplicial sets admits both adjoints, providing left and right adjoints to the inclusion of discrete simplicial spaces in bisimplicial sets:

$$\begin{array}{ccc} \text{Set} & \begin{array}{c} \xleftarrow{\pi_0} \\ \perp \\ \xrightarrow{\text{ev}_0} \end{array} & \text{sSet} & \rightsquigarrow & \text{Set}^{\Delta^{\text{op}}} & \begin{array}{c} \xleftarrow{(\pi_0)_*} \\ \perp \\ \xrightarrow{\text{ev}_{-,0}} \end{array} & \text{sSet}^{\Delta^{\text{op}}} \\ & & & & \text{disc} & & \end{array}$$

Hence the subcategory of discrete simplicial spaces is closed under all limits and colimits, which tells us that all the cubes, simplices, and more general shapes are discrete simplicial spaces. In particular, the conclusion of the following lemma applies to all of the shapes in the simplicial type theory and the functions between them.

Lemma A.6. *Any map of discrete bisimplicial sets is a Reedy fibration. In particular, any discrete simplicial space is Reedy fibrant.*

Proof. First note that any map of discrete simplicial sets is a Kan fibration, for if $S \rightarrow T$ is a map of discrete simplicial sets then the displayed lifting problems are transposes:

$$\begin{array}{ccc} \Lambda_k^n & \longrightarrow & S \\ \downarrow & \nearrow & \downarrow \\ \Delta^n & \longrightarrow & T \end{array} \rightsquigarrow \begin{array}{ccc} \pi_0 \Lambda_k^n & \longrightarrow & S \\ \parallel & \nearrow \exists! & \downarrow \\ \pi_0 \Delta^n & \longrightarrow & T \end{array}$$

Now the discretely embedded subcategory $\text{Set}^{\Delta^{\text{op}}} \hookrightarrow \text{sSet}^{\Delta^{\text{op}}}$ is reflective and coreflective and thus closed under weighted limits with any weight $W \in \text{Set}^{\Delta^{\text{op}}}$. If $X \rightarrow Y$ is a map of discrete

bisimplicial sets, then

$$\{\Delta^m, X\} \rightarrow \{\partial\Delta^m, X\} \times_{\{\partial\Delta^m, Y\}} \{\Delta^m, Y\}$$

is a map of discrete simplicial sets, and thus is a Kan fibration. \square

In particular, the objects I, Φ, Γ, A in the semantic model are Reedy fibrant objects. This is not necessary for us here, but in other situations it can be useful to know.

We now describe the structure on a comprehension category with shapes that corresponds to our type theory with shapes from §2. No additional structure is required on \mathcal{C}_0 ; the finite products that encode context extension are also sufficient to model product cubes. On \mathcal{C}_1 we require:

Definition A.7. We say that a comprehension category with shapes has **pseudo-stable coherent tope logic** if

- the fibers of \mathcal{C}_1 are preorders that are equivalent to distributive lattices,
- with meets and joins preserved up to isomorphism by reindexing,
- in which reindexings along diagonal maps in \mathcal{C}_0 have left adjoints satisfying the Beck-Chevalley condition, and
- moreover the analogue of (2.2) holds.¹³

The rules in Figure 2 ensure that Definition A.7 holds for the syntactic model; the connectives $\top, \wedge, \perp, \vee$ give the distributive lattice structure. The left adjoint to reindexing along $I \rightarrow I \times I$ takes $t : I \vdash \phi$ **tope** to $t : I, s : I \vdash (t \equiv s) \wedge \phi$ **tope**; that this corresponds to the usual rules of equality is an observation of Lawvere [14]; see also [7].

In the semantic model, since the category of simplicial sets is coherent and is closed in bisimplicial sets under all conical limits and colimits, its subobject posets are distributive lattices with meets and joins preserved by pullback, and left adjoints to pullback of monomorphisms along any monomorphism (such as a diagonal map) are given by composition.

Remark A.8. Definition A.7 is called “pseudo-stable” because the meets and joins in fibers of \mathcal{C}_1 are preserved up to isomorphism by reindexing. This accords with the terminology of [18], although they consider mainly type constructors whose rules do not suffice to determine them uniquely up to isomorphism, so that such pseudo-stability has to be asserted as a structure. The method of [18] (which we will not describe in detail here) then applies to make such structure *strictly* preserved by reindexing, as needed to model type theory.¹⁴

The structure on \mathcal{C}_2 consists, firstly, of analogues of the usual structure for modeling dependent type theory with Σ, Π , identity types, and so on, as described in [18]. This exists in both the syntactic and the semantic model for the usual reasons in each case.

Secondly, we have compatibility with the coherent logic. As usual in a comprehension category, we write $\mathcal{T}(\Gamma)$ for the fiber of \mathcal{T} over $\Gamma \in \mathcal{C}_2$, and $\Gamma.A \rightarrow \Gamma$ for the image of such an object in $(\mathcal{C}_2)_{\mathcal{C}_1}^2$.

Definition A.9. A comprehension category with shapes and pseudo-stable coherent tope logic has **type eliminations for tope disjunction** if the following hold:

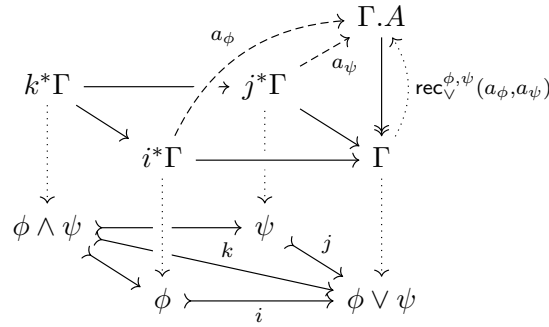
¹³We will not be precise about what (2.2) means categorically, since it holds in the syntactic model by definition, while it holds in the semantic model since there \equiv means literal equality of morphisms.

¹⁴We will not need to consider the more generally “weakly stable” structure of [18], since all the additional operations of our type theory correspond categorically to objects with a universal property that determines them up to isomorphism.

- If $\pi_2(\Gamma)$ is the bottom element of its fiber in \mathcal{C}_1 , then Γ is an initial object of \mathcal{C}_2 .
- If $\pi_2(\Gamma) = \phi \vee \psi$ in a fiber of \mathcal{C}_1 , with injections $i : \phi \rightarrow \phi \vee \psi$ and $j : \psi \rightarrow \phi \vee \psi$ and $k : \phi \wedge \psi \rightarrow \phi \vee \psi$, then the following square of reindexings is a pushout in \mathcal{C}_2 :

$$\begin{array}{ccc} k^*\Gamma & \longrightarrow & i^*\Gamma \\ \downarrow & & \downarrow \\ j^*\Gamma & \longrightarrow & \Gamma \end{array}$$

This appears somewhat different from the rules of Figure 3, which talk about terms $\Gamma \vdash a : A$, hence sections of a comprehension $\Gamma.A \rightarrow \Gamma$. But if Definition A.9 holds then we can define such sections using the universal property of a pushout as in the following diagram.¹⁵



Conversely, Definition A.9 holds in the syntactic model since morphisms of contexts are tuples of sections of dependent types, so a universal property relating to the latter implies one relating to the former.

For the semantic model, the first condition in Definition A.9 is easy since bottom elements of subobject lattices are initial objects, and initial objects in \mathbf{ssSet} are strict (i.e. any map with initial codomain has initial domain). The second condition similarly follows from the facts that in a coherent category, unions of subobjects are pushouts under their intersections, and such pushouts are preserved by pullback, and the inclusion of discrete bisimplicial sets preserves colimits.

Finally, there are the extension types.

Definition A.10. A comprehension category with shapes has **pseudo-stable extension types** if whenever we have the following data:

$$\begin{array}{ccccc} & & (\psi^*\Gamma).A & & \\ & \nearrow a & \downarrow & & \\ \phi^*\Gamma & \longrightarrow & \psi^*\Gamma & \longrightarrow & \Gamma \\ \vdots & & \vdots & & \vdots \\ \Phi \times \phi & \longrightarrow & \Phi \times \psi & \longrightarrow & \Phi \\ & \searrow & \vdots & & \vdots \\ & & \Xi \times I & \longrightarrow & \Xi \end{array} \quad \begin{array}{l} \in \mathcal{C}_2 \\ \downarrow \pi_2 \\ \in \mathcal{C}_1 \\ \downarrow \pi_1 \\ \in \mathcal{C}_0 \end{array} \quad (\text{A.11})$$

with $A \in \mathcal{T}(\psi^*\Gamma)$ there exists $\langle \Pi[\psi][A] \Big|_a^\phi \rangle \in \mathcal{T}(\Gamma)$ whose comprehension $\Gamma. \langle \Pi[\psi][A] \Big|_a^\phi \rangle \rightarrow \Gamma$ represents the functor $(\mathcal{C}_2/\Gamma)^{\text{op}} \rightarrow \mathbf{Set}$ that sends $\sigma : \Theta \rightarrow \Gamma$ to the set of sections b of σ^*A that

¹⁵The vertical dotted arrows denote the action of $\pi_2 : \mathcal{C}_2 \rightarrow \mathcal{C}_1$, rather than an actual morphism in a category, although in the semantic model there is such a morphism.

extend σ^*a , i.e. liftings in the following square:

$$\begin{array}{ccc}
 \phi^*\Theta & \xrightarrow{\sigma^*a} & (\psi^*\Theta).(\sigma^*A) \\
 \downarrow & \dashrightarrow^b & \downarrow \\
 \psi^*\Theta & \xlongequal{\quad} & \psi^*\Theta
 \end{array}
 \tag{A.12}$$

Comparing this to Figure 4, the above diagram of data corresponds exactly to the premises of the first (formation) rule. The second (introduction) rule says that given any b as in Definition A.10 there is an induced map to $\langle \Pi[\psi][A]|\phi_a \rangle$. The third and fourth (elimination) rules say that $\langle \Pi[\psi][A]|\phi_a \rangle$ comes with a universal such b , and the fifth and sixth (β -reduction and η -conversion) rules say that any b is induced by the universal one and that the corresponding map is uniquely determined. In particular, the syntactic model has pseudo-stable extension types.

The fact that the semantic model also has pseudo-stable extension types is the least trivial part of the semantics. Although our primary interest is in the bisimplicial sets model, it is hardly any more work to prove a more general theorem.

Let \mathfrak{T} be a propositional coherent theory, i.e. a set of axiomatic cubes, cube terms, topes, and tope entailments in the first two layers of our type theory from §2, such as the simplicial type theory of §3.1 or the cubical type theory mentioned in Remark 3.2. This gives rise to a syntactic fibration $\mathfrak{T}_1 \rightarrow \mathfrak{T}_0$ as in the first two layers of a comprehension category with shapes, which has coherent tope logic. A *model* of \mathfrak{T} in a topos (or more generally a coherent category) \mathcal{V} is a morphism of fibrations from this syntactic one to the fibration $\text{Mono}(\mathcal{V}) \rightarrow \mathcal{V}$ which preserves finite products in the base and the lattice structure in the fibers.

Definition A.13. A **model category with \mathfrak{T} -shapes** consists of:

- A right proper Cisinski model category \mathcal{M} , i.e. a right proper cofibrantly generated model structure on a Grothendieck topos whose cofibrations are the monomorphisms;
- A model of \mathfrak{T} in a coherent category \mathcal{V} ;
- A coherent functor $\varpi : \mathcal{V} \rightarrow \mathcal{M}$;
- Such that for any object $U \in \mathcal{V}$, the functor $(\varpi U \times -) : \mathcal{M} \rightarrow \mathcal{M}$ preserves acyclicity of cofibrations.

Note that since ϖ preserves finite limits, it preserves monomorphisms, i.e. it takes them to cofibrations in \mathcal{M} . Thus, essentially the same proof of Lemma A.4 implies that for any monomorphism $i : U \rightarrow V$ in \mathcal{V} and any fibration $p : X \rightarrow Y$ in \mathcal{M} , the induced map $\widehat{\{i, p\}} : X^{\varpi V} \rightarrow X^{\varpi U} \times_{Y^{\varpi U}} Y^{\varpi V}$ is a fibration, which is acyclic if p is.

Our primary class of examples is the following.

Example A.14. Let $\mathcal{V} = \mathbf{Set}^{I^{\text{op}}}$ be a presheaf topos containing a model of \mathfrak{T} . For instance, it might be the classifying topos of \mathfrak{T} , if that happens to be a presheaf topos. Let \mathcal{N} be a right proper Cisinski model category, and give $\mathcal{M} = \mathcal{N}^{I^{\text{op}}}$ with the injective model structure, with cofibrations and weak equivalences levelwise; note \mathcal{M} is again a right proper Cisinski model category. Let $\varpi : \mathcal{V} = \mathbf{Set}^{I^{\text{op}}} \rightarrow \mathcal{N}^{I^{\text{op}}} = \mathcal{M}$ be induced by the unique cocontinuous functor $\mathbf{Set} \rightarrow \mathcal{N}$, which takes a set U to the coproduct $\coprod_U 1$ of that many copies of the terminal object. This is a coherent functor since it is the inverse image of a geometric morphism $\mathcal{M} \rightarrow \mathcal{V}$. The final condition follows since the cartesian product in $\mathcal{M} = \mathcal{N}^{I^{\text{op}}}$ is levelwise, as are its acyclic cofibrations, and acyclic cofibrations are closed under \mathbf{Set} -indexed copowers in any model category.

In particular, taking $I = \Delta$ with the universal strict interval in $\mathbf{Set}^{I^{\text{op}}} = \mathbf{sSet}$, and $\mathcal{N} = \mathbf{sSet}$ with the Quillen model structure, we recover the bisimplicial sets model considered above.¹⁶ More generally, we can take $I = \Delta$ with the same universal strict interval, but \mathcal{N} any right proper Cisinski model category; this yields a synthetic theory of “internal $(\infty, 1)$ -categories in \mathcal{N} ”.

However, the following class of examples is also somewhat interesting.

Example A.15. Let $\mathcal{V} = \mathbf{sSet}$ with the universal strict interval, and let $\mathcal{M} = \mathbf{sSet}^{J^{\text{op}}}$ be a topos of simplicial presheaves, with some left Bousfield localization of the injective model structure associated to the Quillen model structure, and assume that \mathcal{M} is right proper and a simplicial model category. Any locally cartesian closed locally presentable $(\infty, 1)$ -category, such as a Grothendieck $(\infty, 1)$ -topos, can be presented by such a model category \mathcal{M} . Let ϖ be restriction along the projection $\Delta \times J \rightarrow \Delta$. Then for $U \in \mathcal{V}$ and $X \in \mathcal{M}$, the product $\varpi U \times X$ is equivalently the simplicial copower of the simplicial enrichment of \mathcal{M} , so the final condition in Definition A.13 follows from the axioms of a simplicial model category.

In the resulting model of simplicial type theory, all types are “discrete” in the sense of §7, since A^2 is just the simplicial path-object. However, this is not completely pointless, since compared to the identity types of ordinary homotopy type theory, the hom-types $\text{hom}_A(x, y)$ of simplicial type theory have strictly functorial behavior, yielding some (but not all) of the advantages of cubical type theory.

From any model category with \mathfrak{T} -shapes, we construct a comprehension category with shapes as follows.

- $\mathcal{C}_0 = \mathcal{V}$ and $\mathcal{C}_1 = \text{Mono}(\mathcal{V})$.¹⁷
- \mathcal{C}_2 is the category of diagrams $\Gamma \rightarrow \varpi\Phi \rightarrow \varpi I$ where $\Phi \rightarrow I$ is an object of \mathcal{C}_1 and $\Gamma \rightarrow \varpi\Phi$ is any fibration in \mathcal{M} .
- \mathcal{T} is the category of diagrams $A \twoheadrightarrow \Gamma \twoheadrightarrow \varpi\Phi \rightarrow \varpi I$, where $\Gamma \twoheadrightarrow \varpi\Phi \rightarrow \varpi I$ is as in \mathcal{C}_2 and $A \twoheadrightarrow \Gamma$ is a fibration in \mathcal{M} .

For the same reasons described above for the bisimplicial set model, this comprehension category has pseudo-stable coherent tope logic with type eliminations for tope disjunction. The latter uses the fact that ϖ is a coherent functor. It remains to prove:

Theorem A.16. *For any model category with \mathfrak{T} -shapes, the above comprehension category with shapes has pseudo-stable extension types.*

Proof. A shape inclusion $t : I \mid \phi \vdash \psi$ is modeled by a monomorphism $i : \phi \rightarrow \psi$ in \mathcal{V} . In the case where A and a are defined in the empty context, then the extension type $\langle \Pi[\psi][A] \mid_a^\phi \rangle$ is constructed simply by the pullback:

$$\begin{array}{ccc} \langle \Pi[\psi][A] \mid_a^\phi \rangle & \longrightarrow & A^{\varpi\psi} \\ \downarrow & \lrcorner & \downarrow A^{\varpi i} \\ 1 & \xrightarrow{a} & A^{\varpi\phi} \end{array}$$

¹⁶Since Δ is an elegant Reedy category in the sense of [4], the Reedy model structure on $\mathcal{N}^{\Delta^{\text{op}}}$ coincides with the injective model structure.

¹⁷In the bisimplicial sets model, we restricted to the subcategory of \mathcal{V} consisting of cubes, but this is immaterial since the image of the interpretation functor from the syntactic model will automatically land in that subcategory anyway.

where $A^{\varpi i}$ is a fibration by the observation after Definition A.13. In the general case, extension types are again constructed by a similar pullback, though there is some delicacy in expressing the context dependence correctly.

Recall that the context $\Xi \mid \Phi \vdash \Gamma \text{ ctx}$ is modeled by a fibration $\Gamma \twoheadrightarrow \varpi\Phi$, where $\Phi \twoheadrightarrow \Xi$ is another monomorphism in \mathcal{V} . The type A is then a further fibration $p: A \twoheadrightarrow \Gamma \times \varpi\psi$, and the dependent term $a: A$ is a section as shown in the diagram below, most of which is just (A.11) specialized to the model in question. For brevity we omit ϖ from the notation, identifying objects and morphisms in \mathcal{V} with their images in \mathcal{M} . Importantly, note that the exponentials are in \mathcal{M} and not in any slice category thereof.

$$\begin{array}{ccccc}
 & & A & & \langle \Pi[\psi][A]_a^\phi \rangle & \longrightarrow & A^\psi \\
 & & \downarrow p & & \downarrow & \lrcorner & \downarrow \widehat{\{i,p\}} \\
 \Gamma \times \phi & \xrightarrow{\Gamma \times i} & \Gamma \times \psi & \xrightarrow{\pi} & \Gamma & \xrightarrow{(a,\eta)} & A^\phi \times_{(\Gamma \times \psi)^\phi} (\Gamma \times \psi)^\psi \\
 \downarrow & \lrcorner & \downarrow & \lrcorner & \downarrow & & \\
 (\Phi \times \phi) & \xrightarrow{(\Phi \times i)} & (\Phi \times \psi) & \xrightarrow{\pi} & \Phi & & \\
 & \searrow & \downarrow & & \downarrow & & \\
 & & (\Xi \times I) & \longrightarrow & \Xi & &
 \end{array}$$

The extension type $\langle \Pi[\psi][A]_a^\phi \rangle$ is constructed using the pullback square in the upper right of this diagram. By the observation after Definition A.13, the map $\widehat{\{i,p\}}$ is a fibration. The map $a: \Gamma \rightarrow A^\phi$ is a transpose of the partial section a of p displayed on the left, while the map $\eta: \Gamma \rightarrow (\Gamma \times \psi)^\psi$ is the transpose of the identity. Since $p \circ a = \Gamma \times i$, this pair indeed defines a cone over the pullback $A^\phi \times_{(\Gamma \times \psi)^\phi} (\Gamma \times \psi)^\psi$; thus the pullback is well-formed. Of course, a pullback of a fibration is a fibration, so $\langle \Pi[\psi][A]_a^\phi \rangle$ defines an element of $\mathcal{T}(\Gamma)$. Thus, it remains to argue that it has the correct universal property.

The universal property of $A^\phi \times_{(\Gamma \times \psi)^\phi} (\Gamma \times \psi)^\psi$ is that it classifies commutative squares from $i: \phi \twoheadrightarrow \psi$ to $p: A \twoheadrightarrow \Gamma \times \psi$, and $\widehat{\{i,p\}}$ classifies lifts in such squares. That is, maps from Θ into this pullback correspond to commutative squares of the form

$$\begin{array}{ccc}
 \Theta \times \phi & \longrightarrow & A \\
 \Theta \times i \downarrow & \lrcorner & \downarrow p \\
 \Theta \times \psi & \longrightarrow & \Gamma \times \psi
 \end{array}$$

(i.e. pairs of maps $\Theta \times \phi \rightarrow A$ and $\Theta \times \psi \rightarrow \Gamma \times \psi$ making the square commute), while lifts of such maps along $\widehat{\{i,p\}}$ correspond to diagonal fillers in such squares. The classifying map of such a commutative square factors through (a,η) just when the bottom map is of the form $\sigma \times 1_\psi$ and the top of the form $a \circ (\sigma \times 1_\phi)$ for some $\sigma: \Theta \rightarrow \Gamma$. Thus, lifts of a given $\sigma: \Theta \rightarrow \Gamma$ to $\langle \Pi[\psi][A]_a^\phi \rangle$ classify lifts in the square

$$\begin{array}{ccc}
 \Theta \times \phi & \xrightarrow{a \circ (\sigma \times 1_\phi)} & A \\
 \Theta \times i \downarrow & \lrcorner & \downarrow p \\
 \Theta \times \psi & \xrightarrow{\sigma \times 1_\psi} & \Gamma \times \psi
 \end{array}$$

Factoring such a square through the pullback of p along $\sigma \times 1_\psi$, we obtain exactly (A.12). \square

This proof is the semantic reason for requiring the shape inclusion $i : I \mid \phi \vdash \psi$ in Figure 4 to be defined in the empty context rather than allowed to depend on Ξ and Φ as well. Specifically, Lemma A.4 and its generalizations do *not* extend to exponentials in slice categories, so if we allowed such dependence then the analogue of $\widehat{\{i, p\}}$ would not necessarily be a fibration.

We also require:

Theorem A.17. *The pseudo-stable extension types in any model category with \mathfrak{T} -shapes satisfy relative function extensionality.*

Proof. A fibration has “contractible fibers” in type theory just when it is an acyclic fibration model-categorically. Thus, Axiom 4.6 holds for the same reason that $\langle \Pi[\psi][A] \mid_a^\phi \rangle \rightarrow \Gamma$ is a fibration, since $\widehat{\{i, p\}}$ is acyclic if p is. \square

In conclusion, we have:

Theorem A.18. *The comprehension category with shapes constructed from any model category with \mathfrak{T} -shapes has pseudo-stable coherent tope logic with type eliminations for tope disjunction, and also pseudo-stable extension types satisfying relative function extensionality.*

Thus, by applying the coherence methods of [18], we can construct from it a strict comprehension category with shapes having strictly stable coherent tope logic with type eliminations for tope disjunction and strictly stable extension types satisfying relative function extensionality. An initiality theorem will then imply that the syntactic model maps into it uniquely, thereby interpreting our type theory with shapes into any model category with shapes, and in particular into bisimplicial sets.

Remark A.19. A comprehension category with shapes does *not* necessarily have “universe types”, and in a general model category with shapes there is no obvious way to construct these. In [27] it was shown that the Reedy model structure on bisimplicial sets *does* have the requisite structure to model universe types, which moreover satisfy the univalence axiom, and those universes carry over to our type theory with shapes. So in the case of the primary motivating model there is no additional difficulty here, but in the cases of Examples A.14 and A.15 there may not be universes. However, as noted in Remark 2.5, in this paper we did not really use the universe in any essential way; so at the expense of a bit more cumbersome notation our results apply just as well to these examples.

In the general case, the axioms of the theory \mathfrak{T} are satisfied by assumption. For the theory of simplices and the bisimplicial set model, by construction we have $2 := \Delta^1 \square \Delta^0$ as the categorically-embedded 1-simplex, with $0, 1 : 2$ the elements of $2_{0,0}$ corresponding, as usual, to the 1st and 0th face maps. The discretely embedded inclusion $\Delta^2 \rightarrow \Delta^1 \times \Delta^1$ models the inequality tope $t : 2, s : 2 \vdash (t \leq s)$ tope. The fact that this satisfies the theory of a strict interval is part of the theorem, mentioned in §3, that simplicial sets are the classifying topos of that theory.

A.3 Segal spaces and Rezk spaces In this section we show that the Segal types of §5 correspond exactly to the Segal spaces in the bisimplicial sets model. A similar argument proves that the Rezk types of §10 also correspond to the Rezk spaces.

Definition A.20. A Reedy fibrant bisimplicial set X is a **Segal space** if and only if for all $m \geq 2$ and $0 < i < m$ the induced map

$$\{\Delta^m, X\} \rightarrow \{\Lambda_i^m, X\}$$

on weighted limits is a trivial fibration in \mathbf{sSet} .

Proposition A.21. *A Reedy fibrant bisimplicial set X is a Segal space if and only if the induced map*

$$X^{\Delta^2 \square \Delta^0} \rightarrow X^{\Lambda_1^2 \square \Delta^0} \tag{A.22}$$

is a Reedy trivial fibration.

Proof. Transposing across the adjunction between the cartesian product and internal hom for bisimplicial sets, (A.22) is a Reedy trivial fibration if and only if X has the right lifting property with respect to the set of maps

$$\{((\partial\Delta^m \hookrightarrow \Delta^m) \widehat{\square} (\partial\Delta^n \hookrightarrow \Delta^n)) \widehat{\times} (\Lambda_1^2 \square \Delta^0 \hookrightarrow \Delta^2 \square \Delta^0)\}_{m,n \geq 0}.$$

This set is isomorphic to

$$\{((\partial\Delta^m \hookrightarrow \Delta^m) \widehat{\times} (\Lambda_1^2 \hookrightarrow \Delta^2)) \widehat{\square} (\partial\Delta^n \hookrightarrow \Delta^n)\}_{m,n \geq 0},$$

where the left-hand product is now the cartesian product on \mathbf{sSet} . Transposing across the weighted limit adjunction, we see that (A.22) is a Reedy trivial fibration if and only if the induced map on weighted limits

$$\{\Delta^m \times \Delta^2, X\} \rightarrow \{\Delta^m \times \Lambda_1^2 \bigcup_{\partial\Delta^m \times \Lambda_1^2} \partial\Delta^m \times \Delta^2, X\}.$$

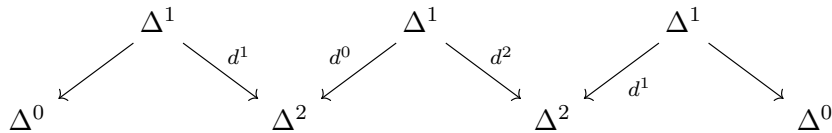
is a trivial fibration of simplicial sets. By the following combinatorial lemma of Joyal, this precisely characterizes the Segal spaces. \square

Lemma A.23 (Joyal [19, 2.3.2.1]). *The following sets generate the same class of morphisms of simplicial sets under coproduct, pushout, retract, and sequential composition:*

1. *The inner horn inclusions $\Lambda_i^m \hookrightarrow \Delta^m$ for $m \geq 2$, $0 < i < m$.*
2. *The collection of all inclusions*

$$\{\Delta^m \times \Lambda_1^2 \bigcup_{\partial\Delta^m \times \Lambda_1^2} \partial\Delta^m \times \Delta^2 \hookrightarrow \Delta^m \times \Delta^2\}_{m \geq 0}.$$

Let E denote the simplicial set defined as the colimit of the diagram



The simplicial set E together with its “middle” 1-simplex may be regarded as the “free-living bi-invertible map”, equipped with left and right inverses.

Definition A.24. A Segal space X is a **Rezk space** if and only if the map

$$\{E, X\} \rightarrow \{\Delta^0, X\} \cong X_0$$

on weighted limits induced by either vertex map $\Delta^0 \rightarrow E$ is a trivial fibration in \mathbf{sSet} .

Our first task is to re-express the Rezk-completeness condition in the internal language of bisimplicial sets.

Proposition A.25. *A Segal space X is a Rezk space if and only if the induced map*

$$X^{E\Box\Delta^0} \rightarrow X^{\Delta^0\Box\Delta^0} \cong X \tag{A.26}$$

is a Reedy trivial fibration.

Proof. As in the proof of Proposition A.21, the condition that (A.26) is a Reedy trivial fibration transposes to the condition that the induced map on weighted limits

$$\{\Delta^m \times E, X\} \rightarrow \{\Delta^m \times \Delta^0 \bigcup_{\partial\Delta^m \times \Delta^0} \partial\Delta^m \times E, X\}$$

is a trivial fibration of simplicial sets for all $m \geq 0$. In the case $m = 0$ this is the condition of Definition A.24 so we see that the lifting property (A.26) implies the completeness condition.

For the converse we appeal to known model categorical results to avoid having to prove a combinatorial lemma. The inclusion $\Delta^0 \rightarrow E$ is a trivial cofibration in the Joyal model structure and the discrete embedding $-\Box\Delta^0: \mathbf{sSet} \rightarrow \mathbf{ssSet}$ of simplicial sets into bisimplicial sets is a left Quillen equivalence from the Joyal model structure to the Rezk model structure [11, 4.11]. As the Rezk model structure is cartesian monoidal with the Rezk spaces as its fibrant objects, it follows that if X is a Rezk space, then the map (A.26) is a trivial fibration. \square

Proposition A.25 is equivalent to the condition that the map $X \rightarrow X^{E\Box\Delta^0}$ is an equivalence in the Rezk model structure or equivalently in the Reedy model structure. Here $X^{E\Box\Delta^0}$ is a model for the total space of the type of isomorphisms introduced in §10.1. Thus, Proposition A.25 corresponds to the Rezk-completeness condition of Definition 10.6.

Remark A.27. There is also a similar characterization of our covariant fibrations. By Theorem 8.5, a Reedy fibration $\pi: C \rightarrow A$ of bisimplicial sets is a covariant fibration in our sense if and only if the square

$$\begin{array}{ccc} C^2 & \xrightarrow{\pi^2} & A^2 \\ \text{ev}_0 \downarrow & & \downarrow \text{ev}_0 \\ C & \xrightarrow{\pi} & A \end{array} \tag{A.28}$$

is a homotopy pullback. If A and C are Rezk, then this corresponds exactly to the characterization of “groupoidal cartesian fibrations” in [25, Proposition 4.2.7], specialized to the ∞ -cosmos of Rezk spaces.

On the other hand, for arbitrary A and C , Lemma A.4 guarantees that the induced map to the pullback

$$\langle C^{i_0}, \pi^2 \rangle: C^2 \rightarrow C \times_A A^2 \tag{A.29}$$

is a Reedy fibration; thus (A.28) is a homotopy pullback square if and only if (A.29) is a Reedy trivial fibration. By [13, Lemma 2.1.3], this happens if and only if for each $n \geq 1$ the square of simplicial sets

$$\begin{array}{ccc} C_n & \xrightarrow{\pi} & A_n \\ \text{ev}_0 \downarrow & & \downarrow \text{ev}_0 \\ C_0 & \xrightarrow{\pi} & A_0 \end{array}$$

is a homotopy pullback; such maps are called *left fibrations* in [13]. By [6, Proposition 1.7] and [21, Lemma 3.9], if A and C are Segal spaces then it suffices to assert this for $n = 1$. Moreover, by [6, Proposition 1.10], left fibrations over a Segal space A are the fibrant objects in a model structure

that is Quillen equivalent to left fibrations over quasi-categories, and by [21, Theorem 4.8] this remains true for arbitrary A . Thus, just as our Rezk types coincide with Rezk spaces and hence model $(\infty, 1)$ -categories, our covariant fibrations model ∞ -groupoid-valued $(\infty, 1)$ -functors.

Acknowledgements

This material is based on research sponsored by The United States Air Force Research Laboratory under agreement number FA9550-15-1-0053. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the United States Air Force Research Laboratory, the U.S. Government, or Carnegie Mellon University. The first-named author is also grateful for support from the National Science Foundation via grants DMS-1551129 and DMS-1619569, the latter supporting a conference after which some of this work took place. A valuable discussion of the results of this paper took place at the 2017 Mathematics Research Community in Homotopy Type Theory coordinated by the AMS and supported by the National Science Foundation under grant number DMS-1321794. The authors wish to thank the anonymous referee for a lengthy list of cogent suggestions and Arthur Azevedo de Amorim for catching a number of typos and imprecisions in the syntax for our type theory with shapes.

References

- [1] Benedikt Ahrens, Krzysztof Kapulkin, and Michael Shulman. Univalent categories and the Rezk completion. *Mathematical Structures in Computer Science*, 25:1010–1039, 6 2015. arXiv:1303.0584.
- [2] Danil Annenkov, Paolo Capriotti, and Nicolai Kraus. Two-level type theory and applications. arXiv:1705.03307, 2017.
- [3] David Ayala and John Francis. Fibrations of ∞ -categories. arXiv:1702.02681, 2017.
- [4] Julia E. Bergner and Charles Rezk. Reedy categories and the Θ -construction. *Math. Z.*, 274(1-2):499–514, 2013. arXiv:1110.1066.
- [5] Cyril Cohen, Thierry Coquand, Simon Huber, and Anders Mörtberg. Cubical type theory: a constructive interpretation of the univalence axiom. arXiv:1611.02108, 2016.
- [6] Pedro Boavida de Brito. Segal objects and the Grothendieck construction. arXiv:1605.00706, 2016.
- [7] Bart Jacobs. *Categorical Logic and Type Theory*. Number 141 in Studies in Logic and the Foundations of Mathematics. North Holland, Amsterdam, 1999.
- [8] Peter T. Johnstone. On a topological topos. *Proc. London Math. Soc. (3)*, 38(2):237–271, 1979.
- [9] A. Joyal. Quasi-categories and Kan complexes. *Journal of Pure and Applied Algebra*, 175:207–222, 2002.

- [10] André Joyal. Disks, duality, and θ -categories. Draft available at <https://ncatlab.org/nlab/files/JoyalThetaCategories.pdf>, 1997.
- [11] André Joyal and Myles Tierney. Quasi-categories vs Segal spaces. In *Categories in Algebra, Geometry and Mathematical Physics*, pages 277–326. American Mathematical Society, 2006. arXiv:math/0607820.
- [12] Chris Kapulkin and Peter LeFanu Lumsdaine. The simplicial model of univalent foundations (after Voevodsky). arXiv:1211.2851, 2012.
- [13] David Kazhdan and Yakov Varshavsky. Yoneda lemma for complete segal spaces. arXiv:1401.5656, 2014.
- [14] F. William Lawvere. Equality in hyperdoctrines and comprehension schema as an adjoint functor. In *Applications of Categorical Algebra (Proc. Sympos. Pure Math., Vol. XVII, New York, 1968)*, pages 1–14. Amer. Math. Soc., Providence, R.I., 1970.
- [15] F. William Lawvere. Axiomatic cohesion. *Theory and Applications of Categories*, 19(3):41–49, 2007.
- [16] Dan Licata and Robert Harper. 2-dimensional directed dependent type theory. *MFPS*, 2011. <http://www.cs.cmu.edu/~drl/pubs/lh102dtt/lh102dtt.pdf>.
- [17] Peter LeFanu Lumsdaine. Weak omega-categories from intensional type theory. *Typed lambda calculi and applications*, 6:1–19, 2010. arXiv:0812.0409.
- [18] Peter LeFanu Lumsdaine and Michael A. Warren. The local universes model: An overlooked coherence construction for dependent type theories. *ACM Trans. Comput. Logic*, 16(3):23:1–23:31, July 2015. arXiv:1411.1736.
- [19] Jacob Lurie. *Higher topos theory*. Number 170 in Annals of Mathematics Studies. Princeton University Press, 2009. arXiv:math.CT/0608040.
- [20] Saunders Mac Lane and Ieke Moerdijk. *Sheaves in geometry and logic: a first introduction to topos theory*. Universitext. Springer-Verlag, New York, 1994. Corrected reprint of the 1992 edition.
- [21] Nima Rasekh. Yoneda lemma for simplicial spaces. arXiv:1711.03160, 2017.
- [22] Charles Rezk. A model for the homotopy theory of homotopy theory. *Trans. Amer. Math. Soc.*, 353(3):973–1007 (electronic), 2001. arXiv:math.AT/9811037.
- [23] Charles Rezk. A cartesian presentation of weak n -categories. *Geometry & Topology*, 14, 2010. arXiv:0901.3602.
- [24] Emily Riehl and Dominic Verity. Homotopy coherent adjunctions and the formal theory of monads. *Advances in Mathematics*, 286(2):802–888, January 2016.
- [25] Emily Riehl and Dominic Verity. Fibrations and yoneda’s lemma in an ∞ -cosmos. *J. Pure Appl. Algebra*, 221(3):499–564, 2017. arXiv:1506.05500.
- [26] Urs Schreiber. Differential cohomology in a cohesive ∞ -topos. arXiv:1310.7930, 2013.

- [27] Michael Shulman. The univalence axiom for elegant Reedy presheaves. *Homology, Homotopy, and Applications*, 17(2):81–106, 2015. arXiv:1307.6248.
- [28] Michael Shulman. Brouwer’s fixed-point theorem in real-cohesive homotopy type theory. To appear in MSCS. arXiv:1509.07584, 2017.
- [29] Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. <http://homotopytypetheory.org/book/>, first edition, 2013.
- [30] Benno van den Berg and Richard Garner. Types are weak ω -groupoids. *Proceedings of the London Mathematical Society*, 102(2):370–394, 2011.
- [31] Vladimir Voevodsky. A simple type system with two identity types. Draft available at <https://www.math.ias.edu/vladimir/sites/math.ias.edu.vladimir/files/HTS.pdf>, see also <https://ncatlab.org/homotopytypetheory/show/Homotopy+Type+System>, 2013.