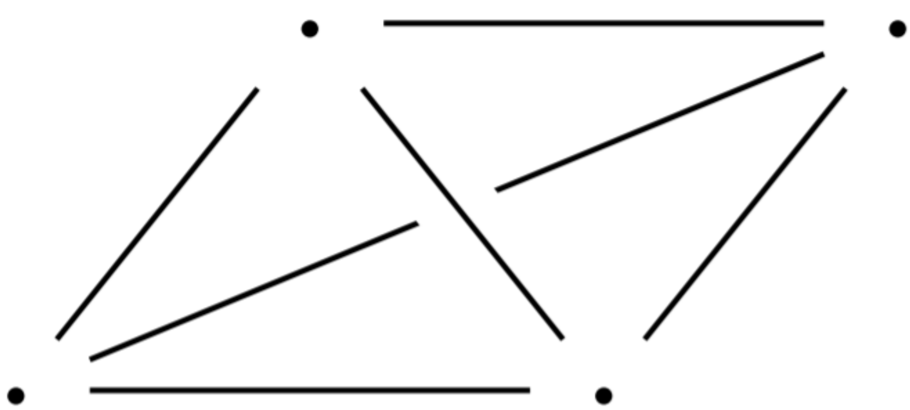
Fredrik Bakke

Segal Spaces in Homotopy Type Theory

Master's thesis in Mathematical Sciences
Supervisor: Rune Haugseng
December 2021



Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

NTNU
Norwegian University of
Science and Technology

Fredrik Bakke

Segal Spaces in Homotopy Type Theory

Master's thesis in Mathematical Sciences
Supervisor: Rune Haugseng
December 2021

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

NTNU
Norwegian University of Science and Technology

Abstract

Homotopy type theory is a foundational language for doing homotopy invariant mathematics, hence one would expect it to be a natural environment in which to study $(\infty, 1)$-categories. However, it is currently an open problem to formulate such objects in this type theory. As an alternative, we may consider an extension of this type theory with a basic notion of strict directed shapes we dub simplicial homotopy type theory. The basic objects of homotopy type theory can be understood as ∞-groupoids. Correspondingly, the basic objects of simplicial homotopy type theory can be understood as simplicial ∞-groupoids. Inside this extension, we have simple axiomatizations of $(\infty, 1)$-precategories and $(\infty, 1)$-categories semantically corresponding to Segal spaces and Rezk spaces, also known as complete Segal spaces. Since all constructions in this theory are homotopy invariant, it presents a natural environment in which to study these objects.

We take a ground up approach and subdivide the thesis into three main parts. First, we study homotopy type theory itself and many of its facets. We then move on to the study of simplicial homotopy theory which is a natural area to interpret the two type theories in. Finally, having laid the proper foundation, we present simplicial homotopy type theory and study $(\infty, 1)$-category theory inside of it. In particular, internalizing certain lifting properties and fibrations.

The original work in this thesis lies in the connecting of ideas, general elaboration on simplicial homotopy type theory, as well as in the internalization of some concepts from $(\infty, 1)$-category theory in classical mathematics into this type theory.

Acknowledgements

I extend my deepest gratitude to my supervisor for giving me the opportunity to write this thesis. It has been an incredible learning experience that would not have been possible without him. His continued guidance and feedback have also been invaluable in shaping this thesis.

I would also like to thank all of the authors who make their amazing work freely available and selflessly contribute to free online resources.

Last but not least, I would like to thank my partner for her unconditional love and support. And her unending patience with me throughout the writing process. Without her, I would surely have gone insane this year.

Fredrik

Contents

Introduction

The thesis is subdivided into three main parts: *homotopy type theory*, *simplicial homotopy theory*, and *simplicial homotopy type theory*.

In the first part we survey homotopy type theory taking a formal approach akin to [UF13, Appendix A.2]. This is a foundational theory of mathematics, posing as an alternative to for instance set theory. In type theory, the basic objects are terms and types, resembling the notions of elements and sets from set theory. We first discuss the philosophy of constructive mathematics and the principle of proof relevance, and its manifestation as "propositions as types." We then lay out the basic syntax and type constructions. After this, we display the weak ∞-groupoid structure carried by types and demonstrate the homotopy invariance of internal constructions. Hence, in this sense *homotopy (type theory)* is a *(homotopy type) theory*. We also visit the concept of higher inductive types, which are inductively defined types with higher-dimensional homotopical structure. In particular, we use the examples of an interval type and circle type as motivation for discussing core concepts of the type theory. Finally, we axiomatize a type of types and discuss the univalence axiom due to Voevodsky [Voe14].

In the second part we survey simplicial homotopy theory. We first define simplicial sets, and consider a range of constructions on them. Simplicial sets have a multitude of uses to us. Firstly, they may be used as an alternative model of the homotopy theory of topological spaces through the Kan–Quillen model structure [Qui67]. And secondly, we may faithfully interpret categories as certain simplicial sets. Hence simplicial sets pose a unifying framework for doing category theory and for doing homotopy theory. So, in particular, they are a good starting point for defining and studying $(\infty, 1)$-categories [Rez01; Joy08; DK80]. We go into particulars on the theory of model categories, but first discuss homotopical categories, the lifting problem, and weak factorization systems taking inspiration from [Rie20]. We apply this to understand the homotopy theory of ∞-groupoids as Kan complexes and the homotopy theory of $(\infty, 1)$-categories as Rezk spaces. We connect this to the semantics of both homotopy type theory [KL18] and later simplicial homotopy type theory [RS17].

Finally, in the third part we study simplicial homotopy type theory as first introduced in [RS17]. We define it as a type theory with three layers. The first two express an intuitionistic first-order logic of strict directed shapes. and on top of this we place the homotopy type theory as laid out in Part I, but with additional types of relative functions from shapes into types. We then establish the basic properties of these relative function types and the simplicial ∞-groupoid structure of types. After this, we study the internal $(\infty, 1)$-category theory. Finally, in the last section, taking inspiration from [BW21] and [RV21], we study certain lifting properties and fibrations in simplicial homotopy type theory.

We assume the reader is well-acquainted with category theory and the homotopy theory of topological spaces. It is also advantageous, but not necessary, for the reader to be familiar with foundational mathematics and enriched category theory.

Part I
Homotopy Type Theory

Homotopy type theory is a recent development in foundational mathematics due to insights connecting Martin–Löf dependent type theory[1] to homotopy theory. It extends this type theory by interpreting types as homotopy spaces and by equipping it with higher inductive types and the univalence axiom.

We will begin by discussing the philosophy of intuitionistic type theory and then present the basic type-theoretic framework. This framework is based on principles that make it distinct in flavor from set theory and classical logic. In particular, we will look at the principle of proof relevance, a manifestation of it: propositions as types, and the philosophy of constructivism.

We will then present a formal view of this type theory, define the basic type formers and discuss their defining rules both in prose and with inference rules. We will take an approach resembling that of [UF13, Appendix A.2], although we do not immediately assume an internal type of types.

Most important to the homotopy interpretation is the *identity* or *equality type*, which houses the paths of a type. Using the identity type, we will define the weak ∞-groupoid structure on types. With this reading all functions may be understood as *continuous* functions.

With this established we shall look at constructions called higher inductive types. These are inductively defined types with generators on their identity types and give a framework for doing synthetic homotopy theory inside the type theory. In particular, we will use the motivating examples of the interval and the circle type to introduce function extensionality and homotopy levels of types.

Finally, we will axiomatize a type of types and discuss the univalence axiom. This axiom has a multitude of interpretations and consequences. For instance, due to the principle of univalence, the formulation of 1-category theory in homotopy type theory is arguably more natural than the set-theoretic counterpart. In contrast to the set-theoretic formulation, the natural form of identification in homotopy type theory is isomorphism, which in many cases is a better-suited form of identification.

1 Proof relevance and constructivity

Intuitionistic type theories are *proof relevant* theories. meaning that whenever a claim of truth is made, it must be accompanied by a proof. Moreover, proofs are first-class citizens and may be manipulated just like any other mathematical object. This is a powerful idea which, in addition to manifesting as *propositions as types*, is also fundamental in the interpretation of types as homotopy spaces.

[1] Also known as *intuitionistic type theory* or *constructive type theory* although these terms are also used to refer to broader classes of type theories.

Unlike set theory, which is built on a logical framework of first-order propositional logic, this correspondence of propositions as types allows us to define and work with propositions directly inside of type theory, hence there is no need for an underlying logic.

The type-theoretic analogues of sets and elements are respectively called **types** and **term**. A proof of the inhabitedness of a type A is simply a term of that type $a : A$. From a logical viewpoint, one would interpret this as proof that A is true. Similarly, a proof of the proposition "A implies B" is a method of constructing a term of B given a term of A, i.e. it is a function term $A \to B$.

It is a remarkable fact that every proposition may be encoded as a type in this way. For instance, to encode the proposition "A is not inhabited" one forms the function type $A \to \mathbf{0}$ where $\mathbf{0}$ denotes the empty type. The only way to construct a function from A to $\mathbf{0}$ is if A is itself empty, in which case there is exactly one such function. Another example, to prove that two terms $x, y : A$ are equal, one must define an element of the identity type $x =_A y$, a type we will define later.

We are purposefully vague with certain ideas in this section, appealing instead to the intuition of the reader. We will talk in terms of *propositions*, as what we discuss specializes to the theory of intuitionistic propositional logic. But one may as well exchange the word "proposition" with "type."

Constructivity

A proof-relevant theory stipulates that the proof itself is relevant to the statement it proves. A claim of truth is accompanied by a proof, and this claim may not be acted upon constructively unless we have access to the proof. For example, a claim of the existence of some object is typically accompanied by a construction of it, and the particular construction itself is relevant to the result. When we want to use the existence of this object in a later statement, we may find use for the specific construction described in the previous proof as well.

In classical logic, on the other hand, proofs are not as relevant to the statement they prove, as they give no other content to it than to demonstrate its *mere validity*. This issue is founded on the fact that classical logic assumes non-constructive axioms. Consider for instance the classical formulation of the axiom of choice.

Axiom 1 (Axiom of choice (AoC))**.** *For every family of merely inhabited*[2] *sets* $\{S_i\}_{i \in I}$ *there exists a choice function*

$$f : \{S_i\}_{i \in I} \to \bigcup_{i \in I} S_i$$

such that $f(S_i) \in S_i$ *for all* $i \in I$.

This axiom gives no method for constructing such a choice function. The axiom postulates only its *mere existence*. Hence the axiom is non-constructive, and we cannot

[2]The notion of "mere inhabitance" of a set coincides with the usual notion of inhabitance in classical logic, but this phrasing alludes to what makes the axiom non-constructive.

extract any computational meaning from it.[3]

Though this axiom is widely accepted, it is not unheard of to question the validity of the axiom of choice. Not just due to its non-constructivity but also because it leads to well-known paradoxes such as the Banach-Tarski paradox. However, as constructivists, we have to go a step further and rid ourselves of the assumption of the law of excluded middle. This boils down to separating the notions of *negation* and *complement* of a proposition.

Definition 1.1 (Negation and complement)**.** Let p be a proposition. The **negation** of p, written $\neg p$ is the proposition

$$\neg p :\equiv p \to \bot$$

where $\bot$ is the trivially false proposition. We can intuitively read $\neg p$ as "assuming p leads to a contradiction." Other readings are "p is refuted" or "p is false" although note that these readings are in terms of provability. The **complement** of p on the other hand is a proposition $\bar{p}$ satisfying

$$p \vee \bar{p} \text{ is true, and } p \wedge \bar{p} \text{ is false.}$$

In constructive logic, such a proposition may not provably exist, and so one has to fall back to the weaker notion of negation in the general case.

Remark 1.2. If the complement $\bar{p}$ exists it is also the negation $\neg p$.

The converse of this statement is not provable in general without the following axiom.

Axiom 2 (Law of excluded middle (LEM))**.** *For every proposition p, we have*

$$p \vee \neg p \textit{ is true.}$$

The law of excluded middle gives the mathematician a very powerful tool, the proof by contradiction: "If one assumes $\neg p$ holds and from it derive a contradiction, then one may conclude that p is true." In logical notation,

Axiom 3 (Double negation law)**.** $\neg\neg p \to p$ *is true.*

Again, like with the axiom of choice, a proof by contradiction is non-constructive. If for instance one proves that some object exists by this kind of proof by contradiction, the rationale is that it must exist because it can't not exist. Hence constructivism requires us to reject both of these axioms. As a particular consequence, there may be propositions for which we cannot deduce either p or $\neg p$, called **undecidable** propositions. We can however say something weaker, namely that we cannot deduce that neither p nor $\neg p$ hold.

Proposition 1.3. *The law of excluded middle is not refuted,*

$$\neg(p \vee \neg p) \textit{ is false.}$$

[3]For constructive mathematics, however, the natural way to prove inhabitedness is by giving a construction of an inhabitant. Formulating the axiom using this notion of inhabitedness, the choice has already been made by hypothesis. As a consequence, a constructive reformulation of the axiom of choice is actually a derivable fact.

In the course of the proof, we perform two proofs of negation, one nested inside the other. Note that the only way to prove $\neg p$ is to assume p and from it derive a contradiction. This does not assume the law of excluded middle, it is just the defining property of negation. What is not allowed is proving that $\neg p$ is false and from it concluding that p is true.

Proof. We begin by assuming $\neg(p \vee \neg p)$ for the sake of reaching a contradiction. To reach a contradiction it suffices to prove $\neg p$. To prove $\neg p$ we must assume p and from it derive a contradiction. If p is true then we in particular have $p \vee \neg p$, which contradicts our initial assumption of $\neg(p \vee \neg p)$. Hence we reached a contradiction and so $\neg p$ must be true. This again contradicts the initial assumption, hence $\neg(p \vee \neg p)$ must be false. □

Digression 1.4 (Modality)**.** From a higher point of view, we may understand double negation as a *modality*, a certain mode of logic, under which we recover classical logic inside of constructive logic: the law of excluded middle does not hold constructively, but it does classically. There are other modalities we can consider inside of Martin–Löf type theory. Specifically, we have one which preserves slightly more information than double negation, the (-1)-truncation. With this modality, we may formulate weak versions, but faithful to the classical axioms, of the law of excluded middle as well as the axiom of choice which we may assume without breaking proof relevance nor consistency with homotopy type theory.

We end this section with a proof of Diaconescu's Theorem, a relevant result from classical logic.

Theorem 1.5 (Diaconescu's Theorem)
The axiom of choice implies the law of excluded middle.

Proof adapted from [Bau13]. Take any proposition p and consider the set $\mathbf{2} = \{0, 1\}$ with the following two subsets

$$A = \{\, x \in \mathbf{2} \mid p \vee (x = 0) \,\} \qquad B = \{\, x \in \mathbf{2} \mid p \vee (x = 1) \,\}.$$

Both subsets are inhabited as $0 \in A$ and $1 \in B$, and their union is $\mathbf{2}$, so we may apply the axiom of choice to get a choice function $f : \{A, B\} \to \mathbf{2}$ with $f(A) \in A$ and $f(B) \in B$. Now let us consider all possible instances of such a choice function. In each case we will prove the proposition in the corresponding cell. The rows represent the value $f(A)$ takes, and the columns represent the value $f(B)$ takes.

	0	1
0	p	$\neg p$
1	$\lightning$	p

(i) For the diagonal entries observe that we have an element of one of the subsets for which $p \vee (x = y)$ is true but $x \neq y$. Hence it must be the case that p is true.

(ii) For the top right case we may prove $\neg p$. If p were to be true, then A and B would consist of the same elements and so $A = B$ by the axiom of set extensionality, but $f(A) \neq f(B)$, a contradiction.

(iii) Lastly, the bottom left instance cannot occur. We would have p by the same argument as for the diagonal. So by the previous argument, we reach a contradiction. $\square$

If you are curious to learn more about constructive mathematics, I can recommend viewing Andrej Bauer's lecture [Bau13], in which he motivates the usage of constructivism and presents a few entertaining albeit less informative aspects of the theory.

2 Syntax

In this section, we will elaborate on the basic syntax we use to present type theory. We will try to build the formalism from the ground up, assuming little background knowledge. We will follow the naming and style conventions of [RS17] and [UF13]. However, for a full treatment of the formal theory, we refer the reader to for instance [SU06], and for a better syntactic presentation of homotopy type theory, we refer the reader to [UF13, App. A].

What is type theory? Type theories are loosely related formal systems in which **terms** and **types** replace the fundamental roles played by elements and sets from set theory. A type theory is commonly presented syntactically as a collection of rules formed out of *judgments*. The judgments needed for an informal presentation of Martin–Löf type theory as in [UF13] are only the following two:

$a : A$	The expression a **is a term of type** A.
$a \equiv a' : A$	The expressions a and a' are **judgmentally equal** as terms of A.

The first judgment expresses the type-theoretic analog of the typical relationship between an element and a set, although we have yet to explain when A is a type. One difference from set theory is that we do not consider a a term unless it may be judged to be a term of a specific type. In set theory on the other hand one usually considers, for instance, the number 1 to live by itself, but also to be a member of the natural numbers, the reals, and so on.

The second judgment expresses a syntactic notion of equality. If terms are judgmentally equal, or *definitionally equal*, it will mean that the language itself does not distinguish between them. This is generally a stricter notion than "a and a' behave the same in every regard."

For our presentation we also want to have syntax for judging expressions to be types and for judging expressions to be equal types.

$A\ \mathsf{type}$	The expression A **is a type**.
$A \equiv A'\ \mathsf{type}$	The expressions A and A' are judgmentally equal as types.

You might wonder how it could have been possible to judge expressions to be types without additional syntax. Without a type judgment, we would have to be able to judge A to be a type with a judgment of the form $A : \mathcal{U}$ for some type $\mathcal{U}$. Meaning that we would need a type of all types. A naive formulation would allow us to judge $\mathcal{U} : \mathcal{U}$. However, as is familiar from naive set theory, this makes the theory inconsistent. Hence the only alternative is to have a larger type $\mathcal{U}'$ in which $\mathcal{U}$ resides. Of course, this only shifts the problem one step over, and so to make this consistent we need an unbounded hierarchy of **universes**

$$A : \mathcal{U}_0 : \mathcal{U}_1 : \mathcal{U}_2 : \ldots : \mathcal{U}_n : \mathcal{U}_{n+1} : \ldots,$$

we need *turtles all the way down*. This means that without type judgments, our logic *needs* to be at least of ω'th-order. We wish to have the opportunity to restrict ourselves to finite orders, hence we need the additional syntax.

Furthermore, although it is possible to develop *informal* homotopy type theory with only the judgments presented, we wish to take a formal approach. In the formal approach, we use a syntax for hypothesized terms, and every judgment is made *in context* of hypotheses. We use the judgments

$$\Gamma\ \mathsf{ctx} \qquad \Gamma \vdash A\ \mathsf{type} \qquad \Gamma \vdash A \equiv A'\ \mathsf{type} \qquad \Gamma \vdash a : A \qquad \Gamma \vdash a \equiv a' : A.$$

In the formal presentation, the rules of the type theory may be formulated as **inference rules**. These are collections of hypothesis judgments τ_i under which we may conclude another judgment τ, denoted

$$\frac{\tau_1 \qquad \ldots \qquad \tau_n}{\tau}.$$

Inference rules are then applied in a formal procedure where we build *derivation trees* by composing and pairing inference rules. The leaves of a derivation tree are its hypotheses and the root yields its conclusion.

As a preliminary example, the following derivation tree is valid in Martin–Löf type theory

$$\cfrac{\cfrac{p : A \times B}{\mathsf{pr}_1\, p : A}(\times\text{-elim})}{\mathsf{in}_1\, \mathsf{pr}_1\, p : A + B}(+\text{-intro}) \quad \text{hence} \quad (p : A \times B) \vdash (\mathsf{in}_1\, \mathsf{pr}_1\, p : A + B).$$

In general, derivation trees may only be finitely deep but are in fact allowed to be infinitely wide.

Alternative presentations. There are alternative ways to present a type theory. One may for instance model type theories using special categories. We will discuss this more in Section 11. This is a well-suited environment for doing general type theory.

Another approach wildly different in flavor is presenting a type theory as a special programming language called a proof assistant. In this presentation, a program written in the language is a collection of judgments, usually complicated by abstractions and special syntax. The successful compilation, also referred to as *type checking*, of such a program is a verification that all judgments are *well formed*, i.e. that they may be derived using the inference rules specified by the language. Hence the implementation of the compiler, or type checker, serves as a specification of the type theory. Instances of such programming languages include Agda, Coq and Lean.[4] Of these, Agda implements a direct extension of Martin–Löf type theory, while Coq and Lean are based on a closely related dependent type theory called the Calculus of Inductive Constructions. In particular, homotopy type theory may be implemented in all three.

3 Martin–Löf type theory

We now define Martin–Löf dependent type theory using the syntax defined in the previous section.

Contexts. We start by defining the contexts. These are dependent lists of variables we call **free terms**. They are free in the sense that their only restriction is the type they inhabit. The definition is an inductive one, using the below inference rules. We note that context juxtaposition is denoted using commas.

$$\frac{}{() \ \mathsf{ctx}}(\text{ctx-empty}) \qquad \frac{\Gamma \vdash A \ \mathsf{type}}{\Gamma, x : A \ \mathsf{ctx}}(\text{ctx-extension}) \qquad \frac{\Gamma, \gamma, \Lambda \ \mathsf{ctx}}{\Gamma, \gamma, \Lambda \vdash \gamma}(\text{ctx-var})$$

The first rule states that we may always form an empty context. The second states that given a well-formed type A in context Γ, we may extend this context with a new **free term** or **free variable** of type A (where we label this term with a symbol that does not appear elsewhere, so as to avoid variable capture). The final rule states that we may conclude any hypothesis of a well-formed context.

Hence a general context is a list of free terms

$$(x_1 : A_1, \dots, x_n : A_n),$$

where each type A_i may depend on the prior variables. We may write A_i as $A_i(x_1, \dots, x_{i-1})$ to emphasize this fact, although we note that this is the default mode of operation.

A basic tool for symbolically manipulating expressions is the substitution operation. Given a free term $x : A$ in context Γ, and an expression τ in the same context, then given some other term $\Gamma \vdash a : A$ we may **substitute** all instances of x in τ with a, denoted $\tau[a/x]$. More generally, for n such terms, we denote the **simultaneous substituion** of all n terms as $\tau[a_1, \dots, a_n/x_1, \dots, x_n]$ and note that this may not be equal to any series

[4] Agda: `wiki.portal.chalmers.se/agda`, Coq: `coq.inria.fr`, Lean: `leanprover.github.io`

of sequential substitutions of these terms, as any of the terms $a_1, \dots, a_n, x_1, \dots, x_n$ may appear as parts of any of the terms $a_1, \dots, a_n$.

In addition we have structural rules of **weakening** and **cut**. However, these rules are admissible: in every instance they are applicable, their conclusion is already derivable (a metatheoretical result). So their explicit assumption is unnecessary, and we decline to elaborate on them.

3.1 Type construction

We define types in the style of natural deduction. We categorize the inference rules for each kind of type into the following five categories.

Formation. Describing when we may form a type of this kind.

Introduction. Describing when we may construct a term of this type.

Elimination. Describing the operations we may perform on terms of this type.

Computation. Describing how the elimination rules relate to the introduction rules.

Uniqueness. Describing how eliminators involving this type are identified uniquely.

Although a type does not need to have rules in all five categories.

3.1.1 Π-types

We begin by defining the **Π-types**, first in prose and then in syntactic terms. These types are also called **dependent function types** or **dependent product types**. This choice of naming will soon become clear. We call their terms **dependent functions** or **Π-terms**.

Π-formation: Given a type A and a family of types $x : A \vdash B\ \mathsf{type}$[5] then we may form the Π-type $\prod_{x:A} B$.

Π-introduction: Given for every $x : A$ a term $b : B$, we may construct the function term $(x \mapsto b) : \prod_{x:A} B$. This is our lambda notation, expressing that x is mapped to b. This rule is sometime refered to as *lambda abstraction* or simply *abstraction*.

Π-elimination: Given a term $f : \prod_{x:A} B$ and a term $a : A$, one may apply f to a to obtain a term $f(a) : B[a/x]$. This rule is also called *application*.

Π-computation: If a dependent function is defined using the lambda expression $x \mapsto b$, then the term we obtain from the elimination rule is b with a substituted for x,

$$(x \mapsto b)(a) \equiv b[a/x].$$

[5] I.e. the type B depends on $x : A$, which we could have written explicitly as $B(x)$.

Π-uniqueness: A dependent function is judgmentally determined by its values

$$f \equiv (x \mapsto f(x)).$$

Now for the syntactic presentation. These are just formalized versions of the previously stated rules, and may be considered equivalent by the reader.

$$\frac{\Gamma \vdash A \ \mathsf{type} \qquad \Gamma, x : A \vdash B \ \mathsf{type}}{\Gamma \vdash \prod_{x:A} B \ \mathsf{type}}(\Pi\text{-form}) \qquad \frac{\Gamma, x : A \vdash b : B}{\Gamma \vdash (x \mapsto b) : \prod_{x:A} B}(\Pi\text{-intro})$$

$$\frac{\Gamma \vdash f : \prod_{x:A} B \qquad \Gamma \vdash a : A}{\Gamma \vdash f(a) : B[a/x]}(\Pi\text{-elim}) \qquad \frac{\Gamma, x : A \vdash b : B \qquad \Gamma \vdash a : A}{\Gamma \vdash (x \mapsto b)(a) \equiv b[a/x] : B[a/x]}(\Pi\text{-comp})$$

$$\frac{\Gamma \vdash f : \prod_{x:A} B}{\Gamma \vdash f \equiv (x \mapsto f(x)) : \prod_{x:A} B}(\Pi\text{-uniq})$$

Figure 1: Dependent product rules

This family of types plays many central roles. Logically, a term of such a type proves "for each x in A, we have $B(x)$" and hence plays the role of the universal quantifier. Set theoretically, the type is the product of a family of types indexed over A. Categorically, they are generalized internal homs. And homotopically, they describe the type of sections of a fibration.[6]

Thanks to the Π-type we do not need to define a non-dependent function type separately, denoted $A \to B$, since we recover them as a special case of Π-types in the case that B does not depend on $x : A$. Non-dependence may be expressed as

$$\frac{\Gamma, x : A \ \mathsf{ctx} \qquad \Gamma \vdash B \ \mathsf{type}}{A \to B \ \mathsf{type}}$$

Example 3.1 (Examples of function terms)**.** The most basic example of a Π-term is the identity function

$$\mathrm{id}_A :\equiv (x \mapsto x) : A \to A,$$

where we use the relational symbol $:\equiv$ for definition, meaning that we define the left-hand side to be judgmentally equal to what is on the right-hand side. We read it as id_A *is defined to be* $(x \mapsto x)$.

This term may be constructed in any context using the deduction tree

$$\cfrac{\cfrac{\cfrac{\Gamma \vdash A \ \mathsf{type}}{\Gamma, x : A \ \mathsf{ctx}}(\mathsf{ctx\text{-}extension})}{\Gamma, x : A \vdash x : A}(\mathsf{ctx\text{-}var})}{\Gamma \vdash (x \mapsto x) : A \to A}(\Pi\text{-intro}).$$

[6]Although this last fact is not as trivial.

As we can see already in this simplest case, writing out the deduction trees is rather tedious and generally little additional information besides verifying that judgments are well-formed. From now on, we will refrain from writing out the deduction trees in full. Instead, we will only mention specifics of which inference rules are applied in select cases, and trust the reader to infer the particulars of a deduction themselves in general.

We can also construct a curried non-dependent function composition operation of type

$$- \circ - : (B \to C) \to ((A \to B) \to (A \to C))$$

defined as

$$- \circ - :\equiv (g \mapsto (f \mapsto (x \mapsto g(f(x)))))$$

When we have defined binary products, the type signature of $- \circ -$ will be equivalent to the perhaps more familiar $((B \to C) \times (A \to B)) \to (A \to C)$ using the usual product-hom adjunction. So it is mostly a matter of taste how we choose to define it. It has become the norm among type theorists however to prefer curried definitions.

Further conflating these viewpoints we will use the following shorthand for iterated abstraction:

$$- \circ - \equiv (g, f, x \mapsto g(f(x))).$$

And similarly, when evaluating a curried function, say $h : A \to (B \to C)$, we may write $h(a)(b)$ as $h(a, b)$. Lastly, we define our arrow-notation to associate to the right, so the type of $- \circ -$ may be written as

$$(B \to C) \to (A \to B) \to A \to C.$$

The composition operation may be generalized to the dependent case in the following manner. Given type families $x : A \vdash B$ type and $x : A, y : B \vdash C$ type, then we may define the composition operation

$$(g, f, x \mapsto g(x, f(x))) : \left(\prod_{(x:A)} \prod_{(y:B(x))} C(x,y)\right) \to \prod_{(f:\prod_{(x:A)} B(x))} \prod_{(x:A)} C(x, f(x)).$$

Although note that now we need to feed $x : A$ to g as well to keep track of the appropriate codomain.

In the presence of the inference rules for the Π-types, the elimination rule may be formulated internally, meaning we may construct it as a term of a type. For instance, the non-dependent eliminator may be typed as a curried evaluation function

$$\mathsf{eval} :\equiv (f, x \mapsto f(x)) : (A \to B) \to A \to B,$$

which may be read as the following tautology: “to define a function from A to B it suffices to give a function from A to B.” But of course, we are still relying on the elimination principle for function types. This demonstrates a useful fact, however: type eliminators may be internalized as function terms.

In the presence of a universe type $\mathcal{U}$ (as we will formalize in Section 6), function terms also internalize the notion of type families. Namely, the type family $x : A \vdash B\ \mathsf{type}$ is equivalently a function term $B : A \to \mathcal{U}$. Note: even in the absence of a universe type we will abusively denote type families in this way. Note that when we take this viewpoint, substitution instead becomes evaluation.

3.1.2 Σ-types

We now define the **Σ-types**. These are also called **dependent sums** or **types of dependent pairs**. We use the symbol Σ for this class of types, as they may be understood as sums indexed over A.

Σ-formation: Given a type A and a family of types $B : A \to \mathcal{U}$ then we may form the Σ-type $\sum_{x:A} B(x)$.

Σ-introduction: Given a term $a : A$ and a term $b : B(a)$ then we may construct the term $(a, b) : \sum_{x:A} B(x)$.

Σ-elimination: The dependent eliminator for Σ-types states that to construct a dependent function out of a Σ-type

$$f : \prod\nolimits_{(p:\sum_{x:A} B(x))} C$$

one must admit for every $a : A$ and $b : B(a)$ a construction of a term of C. I.e. we need a function term

$$f' : \prod_{(a:A)} \prod_{(b:B(a))} C$$

Hence the elimination rule may be read as stating that the Σ-type is freely generated by the dependent pairs. Writing out the internal eliminator in full, we get

$$\mathsf{ind}_{(\sum_{x:A} B(x))} : \left(\prod_{(a:A)} \prod_{(b:B(a))} C \right) \to \prod_{(p:\sum_{x:A} B(x))} C.$$

Σ-computation: The computation rule states that when a function f is constructed using Σ-elimination as just defined, then applying it to a pair (a, b) yields the judgmental equality

$$f((a, b)) \equiv (f(a))(b).$$

There is no uniqueness rule for the Σ-type.

Under the propositions as types interpretation, a term of $\sum_{x:A} B(x)$ can be read as a proof that there exists an $x : A$ such that $B(x)$ is true. Hence this is our analog of the existential quantifier. Set theoretically, the Σ-type may be interpreted as a disjoint sum indexed over A. Categorically, it generalizes products in the same way that Π-types generalize internal homs. And homotopically, Σ-types describe total spaces.

Digression 3.2 (Induction and recursion)**.** The motivation behind using the symbol ind for the dependent eliminators is because they may be interpreted as principles of induction. Likewise, non-dependent elimination rules may be interpreted as principles of recursion. For types where one expects to see some form of induction or recursion, these principles will in fact coincide with them (up to equivalence). For instance, had we defined the type of natural numbers $\mathbb{N}$, its induction principle would be exactly what you would expect: to construct a dependent map from $\mathbb{N}$ to a type family $P : \mathbb{N} \to \mathcal{U}$ it suffices to provide a base term at $p_0 : P(0)$, and for each $n : \mathbb{N}$, a map $P(n) \to P(n+1)$. Read under the logical interpretation: to prove the predicate $P(n)$ for every natural number n it suffices to prove the base case $n \equiv 0$ and to prove the implication $P(n) \to P(n+1)$.

$$\frac{\Gamma \vdash A \ \mathsf{type} \qquad \Gamma, x : A \vdash B \ \mathsf{type}}{\Gamma \vdash \sum_{x:A} B \ \mathsf{type}}(\Sigma\text{-form})$$

$$\frac{\Gamma, x : A \vdash B \ \mathsf{type} \qquad \Gamma \vdash a : A \qquad \Gamma \vdash b : B[a/x]}{\Gamma \vdash (a,b) : \sum_{x:A} B}(\Sigma\text{-intro})$$

$$\frac{\Gamma, p : \sum_{x:A} B \vdash C \ \mathsf{type} \qquad \Gamma, x : A, y : B \vdash c : C[(x,y)/p] \qquad \Gamma \vdash q : \sum_{x:A} B}{\Gamma \vdash \mathsf{ind}_{(\sum_{x:A} B)}((x, y \mapsto c), q) : C[q/p]}(\Sigma\text{-elim})$$

$$\frac{\Gamma, p : \sum_{x:A} B \vdash C \ \mathsf{type} \qquad \Gamma, x : A, y : B \vdash c : C[(x,y)/p] \qquad \Gamma \vdash a : A \qquad \Gamma \vdash b : B[a/x]}{\Gamma \vdash \mathsf{ind}_{(\sum_{x:A} B)}((x, y \mapsto c), (a,b)) \equiv c[a, b/x, y] : C[(a,b)/p]}(\Sigma\text{-comp})$$

Figure 2: Dependent sum rules

In particular we have projection functions

$$\begin{aligned}
\mathsf{pr}_1 &:\equiv \mathsf{ind}_{(\sum_{x:A} B)}(x, y \mapsto x) : \left(\sum_{x:A} B\right) \to A \\
\mathsf{pr}_2 &:\equiv \mathsf{ind}_{(\sum_{x:A} B)}(x, y \mapsto y) : \prod_{(p:\sum_{x:A} B)} B(\mathsf{pr}_1\, p).
\end{aligned}$$

We have the following central construction characterizing the interaction between Π- and Σ-types:

Theorem 3.3 (Type-theoretic axiom of choice)
Given a type A, a type family $x : A \vdash B$ type and another type family $x : A, y : B \vdash C$ type, then we have a function

$$\mathsf{ttac} : \prod_{x:A} \left(\sum_{y:B(x)} C(x,y) \right) \to \sum_{(f:\prod_{x:A} B(x))} \left(\prod_{x:A} C(x, f(x)) \right),$$

given on Π-terms by $\mathsf{ttac}(f) :\equiv (x \mapsto \mathsf{pr}_1\, f(x), x \mapsto \mathsf{pr}_2\, f(x))$.

Proof. having already defined `ttac` on terms, it remains to verify that everything inhabits the correct types, which we leave as an exercise. □

Under the propositions as types correspondence, this has a notably similar reading to that of the classical axiom of choice: "If there for every $x : A$ exists an inhabitant $y : B(x)$ such that $C(x, y)$, then we have a choice function $f : \prod_{x:A} B(x)$ such that $C(x, f(x))$ for every $x : A$." This does not contradict constructivity since the hypothesis is stronger than the classical formulation. The only way to constructively prove that for every $x : A$ there exists a term $y : B(x)$ such that $C(x, y)$, is to construct such a function. This is in contrast to the classical formulation where the default is to assume *mere* existence. In particular, due to its stronger hypothesis, this theorem does not enable us to prove the usual classical results that depend on the axiom of choice.

Remark 3.4. The implication formulated in the previous theorem may be promoted to an equivalence, in which case it may be interpreted as expressing that products distribute over coproducts.

3.1.3 The empty, unit and boolean type

Let us now define the 0-, 1- and 2-term types, called the empty type $\mathbf{0}$, unit type $\mathbf{1}$ and type of booleans $\mathbf{2}$ respectively. Under the propositions as types interpretation the two types $\mathbf{0}$ and $\mathbf{1}$ play the roles of the trivial falsity and the trivial truth. Categorically, they are the initial and terminal objects. The type $\mathbf{2}$ is also important to us as it allows us to define binary operations on types internally without the need for additional inference rules.

These types' formation rules state that they may be formed in any context. Their introduction rules state that they have the expected number of terms each, and their elimination rules state that we need precisely that many terms in the target A to eliminate out of the corresponding type. Finally, the computation rules tell us that if we mapped a term x in the source type $\mathbf{n}$ to a term a in the target type A, then we recover a judgmentally when evaluating at x.

$$\frac{\Gamma\ \mathsf{ctx}}{\Gamma \vdash \mathbf{0}\ \mathsf{type}}(\mathbf{0}\text{-form}) \qquad \frac{\Gamma, t : \mathbf{0} \vdash A\ \mathsf{type} \qquad \Gamma \vdash n : \mathbf{0}}{\Gamma \vdash \mathsf{ind}_{\mathbf{0}}(n) : A[n/t]}(\mathbf{0}\text{-elim})^{7}$$

$$\frac{\Gamma\ \mathsf{ctx}}{\Gamma \vdash \mathbf{1}\ \mathsf{type}}(\mathbf{1}\text{-form}) \qquad \frac{\Gamma\ \mathsf{ctx}}{\Gamma \vdash 0 : \mathbf{1}}(\mathbf{1}\text{-intro})$$

$$\frac{\Gamma, t : \mathbf{1} \vdash A\ \mathsf{type} \qquad \Gamma \vdash a : A[0/t] \qquad \Gamma \vdash n : \mathbf{1}}{\Gamma \vdash \mathsf{ind}_{\mathbf{1}}(a, n) : A[n/t]}(\mathbf{1}\text{-elim})$$

$$\frac{\Gamma, t : \mathbf{1} \vdash A\ \mathsf{type} \qquad \Gamma \vdash a : A[0/t]}{\Gamma \vdash \mathsf{ind}_{\mathbf{1}}(a, 0) \equiv a}(\mathbf{1}\text{-comp})$$

$$\frac{\Gamma\ \mathsf{ctx}}{\Gamma \vdash \mathbf{2}\ \mathsf{type}}(\mathbf{2}\text{-form}) \qquad \frac{\Gamma\ \mathsf{ctx}}{\Gamma \vdash 0 : \mathbf{2} \qquad \Gamma \vdash 1 : \mathbf{2}}(\mathbf{2}\text{-intro})$$

$$\frac{\Gamma, t : \mathbf{2} \vdash A\ \mathsf{type} \qquad \Gamma \vdash a_0 : A[0/t] \qquad \Gamma \vdash a_1 : A[1/t] \qquad \Gamma \vdash n : \mathbf{2}}{\Gamma \vdash \mathsf{ind}_{\mathbf{2}}(a_1, a_2, n) : A[n/t]}(\mathbf{2}\text{-elim})$$

$$\frac{\Gamma, t : \mathbf{2} \vdash A\ \mathsf{type} \qquad \Gamma \vdash a_0 : A[0/t] \qquad \Gamma \vdash a_1 : A[1/t]}{\Gamma \vdash \mathsf{ind}_{\mathbf{2}}(a_0, a_1, 0) \equiv a_0 \qquad \Gamma \vdash \mathsf{ind}_{\mathbf{2}}(a_0, a_1, 1) \equiv a_1}(\mathbf{2}\text{-comp})$$

Figure 3: Rules for **0**, **1** and **2**

The reader may verify that the rules corresponding to each category indeed encapsulate the intended aspect.

Given a type family $B : \mathbf{2} \to \mathcal{U}$ we recover the binary product $B(0) \times B(1)$ as $\prod_{x:\mathbf{2}} B(x)$, and the binary sum $B(0) + B(1)$ as $\sum_{x:\mathbf{2}} B(x)$. Because of this, we do not need to define these operations separately.

3.1.4 Identity types

Under the propositions as types correspondence, there should be a type that reflects the equality of terms. Hence we introduce a family of types that may presently be understood as only reflecting judgmental equality into the type theory, although as we will shortly discuss this will not continue to hold in homotopy type theory. These are called the **identity types** or (**propositional**) **equality types** or **type of equality proofs** or **path types**.

=-formation: Given terms $x, y : A$, we may form the identity type $x =_A y$.

[7] Encoding "ex falso quod libet."

=-introduction: Given a term $a : A$, we may construct the **reflexivitiy** term, or **constant path at** a, $\mathsf{refl}_a : a =_A a$. This rule expresses that if $a \equiv a'$, then they are also equal in the sense of the identity type.

=-elimination: Assume given a type family $C : \prod_{(x,y:A)}(x =_A y) \to \mathcal{U}$. To construct a dependent function over the identity types

$$f : \prod_{(x,y:A)} \prod_{(p:x=_A y)} C(x,y,p)$$

it suffices to consider the case where y is x and p is refl_x, i.e. that $y \equiv x$ and $p \equiv \mathsf{refl}_x$. Written out in full, we have the following internal induction principle for identity types

$$\mathsf{ind}_{=_A} : \left(\prod_{(x:A)} C(x,x,\mathsf{refl}_x) \right) \to \prod_{(x,y:A)} \prod_{(p:x=_A y)} C(x,y,p).$$

=-computation: In the case that y is x and p is refl_x, one recovers the construction used in the eliminator judgmentally.

$$\mathsf{ind}_{=_A}(c, x, x, \mathsf{refl}_x) \equiv c(x)$$

The elimination principle of identity types may be read as stating that the identity types are freely generated by reflexivities, i.e. constant paths, which is also the case in bare-bones intuitionistic type theory. However, as we shall see in later sections this is not the case in homotopy type theory. Indeed, a recurring theme will be adding axioms regarding the identity types. This is the basis for both higher inductive types and the univalence axiom, but also includes more benign principles like the function extensionality axiom (functions that are equal on evaluations are equal). These axioms cannot be formulated in terms of judgmental equality without breaking either the homotopical interpretation, consistency, or decidability of type checking respectively.

With these additional axioms, the content of the identity types becomes much richer, and the interpretation of identity terms as *paths* becomes fruitful. In this interpretation, we consider terms of a type, e.g. $x, y : A$, as *points*, and terms of the identity type $p, q : x =_A y$ as paths between them.

Moreover, by the formation rule, we may form the iterated identity type $p =_{x=_A y} q$ whose inhabitants are then paths between paths, two-dimensional paths. This type may or may not be inhabited. For instance, we may imagine the type A being shaped like a torus, where the two paths p and q are homotopic, while some third path $r : x =_A y$ is not:

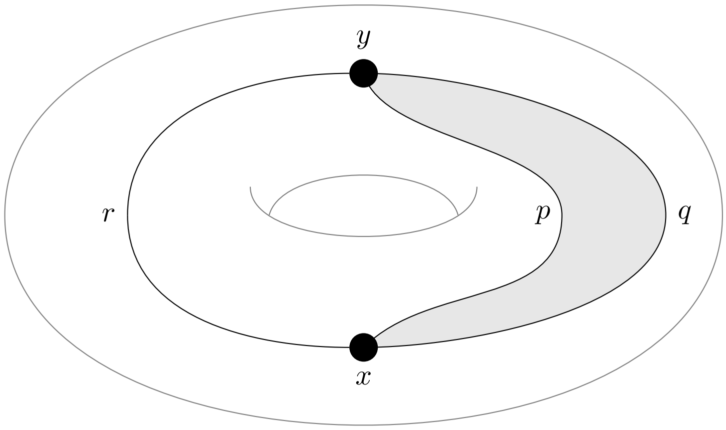


Figure 4: Homotopical interpretation of higher identity terms.

To be precise, we will see that the identity types express a weak groupoid structure of types, and through iterated identity types that types are weak ∞-groupoids. This will be the focus of the next section. For now, we leave you with the formal inference rules of the type family and then discuss some possible alternative extensions of the type theory.

$$\frac{\Gamma \vdash A\ \mathsf{type} \qquad \Gamma \vdash a_1 : A \qquad \Gamma \vdash a_2 : A}{\Gamma \vdash a_1 =_A a_2\ \mathsf{type}}(\text{=-form}) \qquad \frac{\Gamma \vdash A\ \mathsf{type} \qquad \Gamma \vdash a : A}{\Gamma \vdash \mathsf{refl}_a : a =_A a}(\text{=-intro})$$

$$\frac{\begin{array}{c}\Gamma, x_1 : A, x_2 : A, p : x_1 =_A x_2 \vdash C\ \mathsf{type} \\ \Gamma, y : A \vdash c : C[y, y, \mathsf{refl}_y / x_1, x_2, p] \qquad \Gamma \vdash a_1 : A \qquad \Gamma \vdash a_2 : A \qquad \Gamma \vdash q : a_1 =_A a_2\end{array}}{\Gamma \vdash \mathsf{ind}_{=_A}((y \mapsto c), a_1, a_2, q) : C[a_1, a_2, q / x_1, x_2, p]}(\text{=-elim})$$

$$\frac{\Gamma, x_1 : A, x_2 : A, p : x_1 =_A x_2 \vdash C\ \mathsf{type} \qquad \Gamma, y : A \vdash c : C[y, y, \mathsf{refl}_y / x_1, x_2, p] \qquad \Gamma \vdash a : A}{\Gamma \vdash \mathsf{ind}_{=_A}((y \mapsto c), a, a, \mathsf{refl}_a) \equiv c[a/y] : C[a, a, \mathsf{refl}_a / x_1, x_2, p]}(\text{=-comp})$$

Figure 5: Identity rules

3.2 Extensions of Martin–Löf type theory

An important class of types we may include in our type theory are the W-types as is done in [UF13]. These are the types of well-founded trees. They may be compactly characterized as mathematical structures with a well-behaved induction principle. Notable examples of W-types include the natural numbers, the integers, finite sets, and lists. However, these constructions are not needed for our considerations and we omit them from the type theory at present.

We would also like to mention that there are multiple notable extensions of intuitionistic type theory. One has **Homotopy type theory (HoTT)** which extends intensional type theory (ITT) with higher inductive types (HIT) and the univalence axiom (UA)

$$\mathrm{HoTT} = \mathrm{ITT} + \mathrm{HIT} + \mathrm{UA}.$$

Higher inductive types are types which introduce new terms to their path types. For their path types to still make sense, their elimination principles must take into account the new path terms that have been introduced. Univalence is a principle that identifies the equivalence of types with the equality of types. This axiom has many facets, some of which we will visit in a later section. Types act like abstract spaces in homotopy type theory and it gives rise to an intuitionistic theory of weak ∞-groupoids.

Another notable extension is **Extensional type theory (ETT)** which extends intensional type theory with equality of reflection (ER) and uniqueness of identity proofs (UIP)

$$\mathrm{ETT} = \mathrm{ITT} + \mathrm{ER} + \mathrm{UIP}.$$

Equality of reflection states that from an identity term, we may deduce judgmental equality. And uniqueness of identity proofs state that all identity proofs of a given identity type are themselves equal.

Types in this theory may not have higher-dimensional structure due to UIP, so they act like sets. Hence extensional type theory is an intuitionistic theory of sets. However, as a consequence of equality of reflection type checking is undecidable.

As a further extension, we can construct a classical type theory, in which we additionally assume the law of excluded middle and the axiom of choice. This type theory expresses a classical theory of sets which is as strong as Zermelo-Fraenkel set theory with the axiom of choice [Wer97]. However, we lose the benefits of a constructive setting by assuming these axioms *by default*. Instead, when working in an intuitionistic setting we may selectively choose to assume these axioms when needed, while still maintaining the possibility for constructive proofs elsewhere.

In closing, we give a summarizing table taken from [UF13, p. 15], which outlines the correspondences between types, logic, sets and spaces.

Types	Logic	Sets	Homotopy
A type	proposition	set	space
$a : A$	proof	element	point
$B(x)$	predicate	family	fibration
$b(x) : B(x)$	conditional proof	family of elements	section
$\mathbf{0}$, $\mathbf{1}$	$\bot$, $\top$	$\varnothing$, $\{\varnothing\}$	$\varnothing$, $\star$
$A + B$	$A \vee B$	disjoint union	coproduct
$A \times B$	$A \wedge B$	set of pairs	product
$A \to B$	$A \Rightarrow B$	set of functions	function space
$\sum_{(x:A)} B(x)$	$\exists_{(x:A)} B(x)$	disjoint sum	total space
$\prod_{(x:A)} B(x)$	$\forall_{(x:A)} B(x)$	product	space of sections
$x =_A y$	equality	$A \xrightarrow{\Delta} A \times A$	$A^I \twoheadrightarrow A \times A$

Table 1: Summary of basic type, logic, set and homotopy correspondences.

4 Types are ∞-groupoids

In this section, we will demonstrate that the identity type endows types with a weak ∞-groupoid structure. Furthermore, we will see that all constructions in Martin–Löf type theory are *homotopy invariant*: we cannot construct objects internally which do not respect equality.

We begin by making precise the groupoid structure carried by the identity types.

Lemma 4.1. *Given a type A and terms $x, y, z : A$, there is a binary operation*

$$- \cdot - : (x =_A y) \to (y =_A z) \to (x =_A z)$$

called path composition, i.e. the identity relation is transitive.

To familiarize ourselves with arguments applying identity elimination, we are particularly meticulous with this first proof, leaving nothing to the imagination of the reader. After this proof however, we will usually reduce arguments such as this one into single sentences.

Proof. To apply identity elimination we need to construct a dependent map where the terms x, y and z are free:

$$\prod_{x,y,z:A} ((x =_A y) \to (y =_A z) \to (x =_A z)),$$

but we also need to slightly rearrange the arguments:

$$\prod_{x,y:A} \left((x =_A y) \to \prod_{z:A} ((y =_A z) \to (x =_A z)) \right).$$

This type satisfies the hypothesis of the identity eliminator of $(x =_A y)$ which in this case states that to construct a map $\prod_{x,y:A} ((x =_A y) \to C(x,y))$ for some type family

$C : A \to A \to \mathcal{U}$ (since C does not depend on the particular equality proof used), it suffices to define it in the case that y is x and the equality proof $p : x =_A y$ is reflexivity refl_x. I.e. it suffices to define a map $\prod_{x:A} C(x,x)$. In our case, $C(x,y)$ is $(y =_A z) \to (x =_A z)$, and so the problem reduces to defining a function

$$\mathsf{refl}_x \cdot - : (x =_A z) \to (x =_A z).$$

But here the identity function does the trick.

To define $- \cdot -$ as to satisfy the hypothesis of the lemma, where x, y and z are fixed, we may define it as the following abstraction

$$- \cdot - :\equiv p, q \mapsto \mathsf{ind}_{=_A}((r \mapsto r), x, y, p, z, q). \qquad \square$$

Remark 4.2. Our proof of transitivity gives us the computation rule $\mathsf{refl}_x \cdot q \equiv q$. If we had applied the same path induction argument to the second identity type, we would have gotten the computation rule $p \cdot \mathsf{refl}_y \equiv p$. One may also write a proof employing path induction twice, which only admits the computation rule $\mathsf{refl}_x \cdot \mathsf{refl}_x \equiv \mathsf{refl}_x$. This demonstrates an issue with our proof-relevant theory. Since the theorem depends on its proof, one may have to consider not only the theorem but also its proof in applications. And though the statement of the theorem is the same, different proofs may give different computational rules. This is a violation of what they in computer science call the principle of separation of specification and implementation, meaning that one should only need to understand the specification of a program to interface with it. This is a somewhat unfortunate consequence of proof relevance and the usage of computation and uniqueness rules.

Lemma 4.3. *Given a type A and terms $x, y : A$, there is a unary operation*

$$(-)^{-1} : (x =_A y) \to (y =_A x)$$

called path inversion, i.e. the identity relation is symmetric.

Proof. By the equality elimination rule, it suffices to consider the case where y is x and the equality proof of $x =_A y$ is refl_x. But in this case, refl_x will do. $\square$

Theorem 4.4 (Weak groupoid structure of types)
Let A be a type with terms $x, y, z, w : A$ such that p, q and r are terms of appropriate identity types. Then we have the following relations.

$$
\begin{aligned}
p \cdot p^{-1} &=_{(x =_A x)} \mathsf{refl}_x && \text{(Right inverse)}\\
p^{-1} \cdot p &=_{(x =_A x)} \mathsf{refl}_x && \text{(Left inverse)}\\
p \cdot \mathsf{refl}_y &=_{(x =_A y)} p && \text{(Right unit)}\\
\mathsf{refl}_x \cdot p &=_{(x =_A y)} p && \text{(Left unit)}\\
(p \cdot q) \cdot r &=_{(x =_A w)} p \cdot (q \cdot r) && \text{(Associativity)}
\end{aligned}
$$

Proof. We enumerate the cases, and apply path induction in each case.

[RI] By path induction it suffices to consider the case that p is refl_x, in which case we have $\mathsf{refl}_x \cdot \mathsf{refl}_x^{-1} \equiv \mathsf{refl}_x \cdot \mathsf{refl}_x \equiv \mathsf{refl}_x$.

[LI] Again, by path induction it suffices to consider the case that p is refl_x, in which case we have $\mathsf{refl}_x^{-1} \cdot \mathsf{refl}_x \equiv \mathsf{refl}_x \cdot \mathsf{refl}_x \equiv \mathsf{refl}_x$.

[RU] By path induction, it suffices to consider the case that y is x and p is refl_x. But in this case the equality is definitional.

[LU] By Lemma 4.1 this equality is definitional.

[A] Applying path induction it suffices to consider the case where y is x and p is refl_x. But in this case we have $(\mathsf{refl}_x \cdot q) \cdot r \equiv q \cdot r \equiv \mathsf{refl}_x \cdot (q \cdot r)$ by the proof of Lemma 4.1.[8] □

This groupoid structure is *weak* in the sense that equalities stated are generally not reflexivities, i.e. "on the nose," but only hold up to paths one level up. For instance, the following composite of 2-paths witness associativity of path composition:

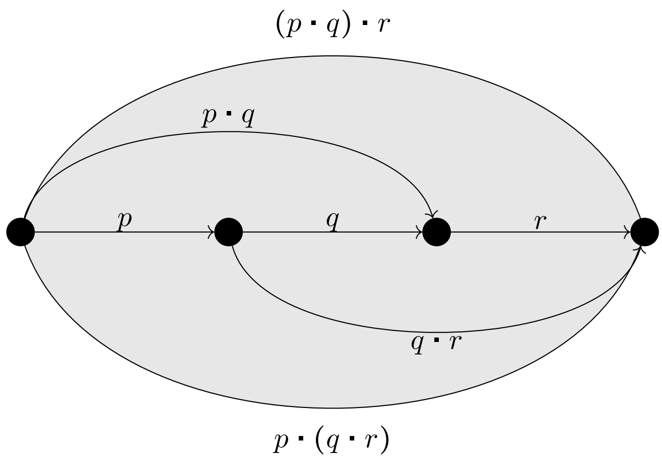


Figure 6: Homotopical interpretation of associativity of path composition, where the gray regions are two-dimensional homotopies.

These higher paths are again subject to the same rules and this pattern continues ad infinitum. Hence there is an infinite hierarchy of weak groupoid structures, called a weak ∞-groupoid structure. Every type is a weak ∞-groupoid. Luckily, we do not need to examine this entire infinite structure on types in every consideration, in fact doing so is impossible internally; we may simply define the level of the structure we need as we go.

Proposition 4.5. $\left(p^{-1}\right)^{-1} =_{(x =_A y)} p$ *and* $(p \cdot q)^{-1} =_{(z =_A x)} q^{-1} \cdot p^{-1}$.

[8]Notice that had we instead proven Lemma 4.1 by double induction as described in Remark 4.2, we would have had to apply path induction thrice to apply the same lemma.

Proof. To prove the first claim it suffices by path induction to consider the case where y is x and p is refl_x in which case by the proof of Lemma 4.3 we have $\left(\mathsf{refl}_x^{-1}\right)^{-1} \equiv \mathsf{refl}_x^{-1} \equiv \mathsf{refl}_x$.

For the second claim, observe that the right-hand side path composition is well-defined. By iterated path induction, it suffices to consider the case where z is y is x and q is p is refl_x, in which case we have definitional equality. □

As a more interesting property of this groupoid structure, we have commutativity of higher-order paths, which we state without proof below.

Theorem 4.6 (Eckmann–Hilton, [UF13, Theorem 2.1.6])
The path composition operation on $\mathsf{refl}_x =_{(x=_A x)} \mathsf{refl}_x$ *is commutative for any* $x : A$.

Even syllepsis, proving that the commutativity proof $p \cdot q = q \cdot p$ emitted by Eckmann–Hilton is inverse to the commutativity proof of $q \cdot p = p \cdot q$, was recently formalized [Soj21].

4.1 Functions are functors

We might hope that maps respect equality, that they are "well-defined" in the classical sense. Of course, we have not put many restrictions on the identity type (except for seemingly stating that it is freely generated by the reflexivities), and one would not expect functions in general to respect every equivalence relation. But still, equality is supposed to be the relation which everything respects.

So suppose we have a map $f : A \to B$. In this simplest case, B will not depend on the argument $x : A$. So we may ask for functions to satisfy the condition that if we have $x, y : A$ and a proof that they are equal $p : x =_A y$, that we can construct a proof that f evaluated at x is equal to f evaluated at y. This is immediately true:

Lemma 4.7 (Action on paths)**.** *Given types A and B we have an **action on paths***

$$\mathsf{ap}_{A,B} : \prod_{(f:A\to B)} \prod_{(x,y:A)} ((x =_A y) \to (f(x) =_B f(y))) \qquad \text{(Action on paths)}$$

Proof. By path induction it suffices to consider the case that y is x and that the path of $x =_A y$ is refl_x. In this case, we have a path $\mathsf{refl}_{f(x)} : (f(x) =_B f(y))$ as desired. □

We will usually assume A, B, x and y implicitly and denote $\mathsf{ap}_{A,B}(f, x, y, p)$ as $\mathsf{ap}_f(p)$. For a function's action on paths, we in fact have functoriality:

Theorem 4.8 (Functoriality of functions)
Given functions $f : A \to B$, $g : B \to C$ and paths $p : x =_A y$, $q : y =_A z$, we have the following equalities:

(i) $\mathsf{ap}_f(p \cdot q) =_{(f(x)=_B f(z))} \mathsf{ap}_f(p) \cdot \mathsf{ap}_f(q)$

(ii) $\mathsf{ap}_f(p^{-1}) =_{(f(x)=_B f(y))} \mathsf{ap}_f(p)^{-1}$

(iii) $\mathsf{ap}_g \circ \mathsf{ap}_f(p) =_{(g(f(x))=_C g(f(y)))} \mathsf{ap}_{g\circ f}(p)$

(iv) $\mathsf{ap}_{\mathrm{id}_A}(p) =_{(x =_A y)} p$.

Notice how under the homotopy interpretation, this theorem states that functions preserve paths, that functions are continuous. But this is under no extra assumptions on them; they are continuous by definition.

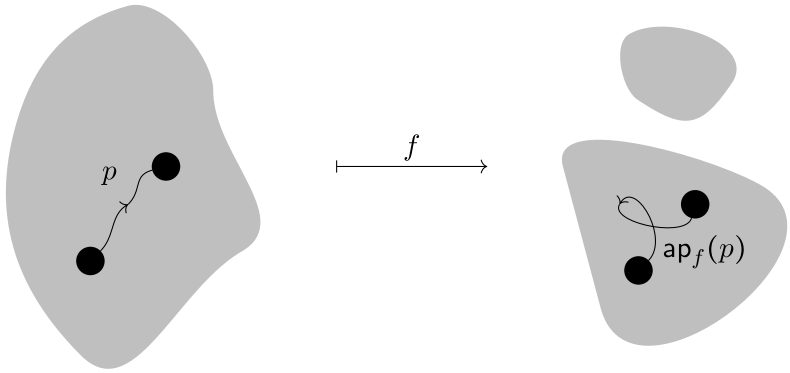


Figure 7: Homotopical interpretation of action on paths. The endpoints of p cannot end up in different path components in the codomain.

Now, if we instead have a dependently typed map $f : \prod_{x:A} P(x)$, the above definition is no longer sufficient. In this case, P may depend on x, so $f(x)$ and $f(y)$ don't necessarily have the same type, meaning we cannot form the identity type $f(x) = f(y)$. If x and y were judgmentally equal, then we would know that the two codomain types are the same by substitution, but we only have that x and y are propositionally equal. Still, we would expect $P(x)$ and $P(y)$ to be related in some way, though we do not go as far as to say the two spaces are the same. We formulate this relation in the following lemma:

Lemma 4.9 (Transport)**.** *Suppose that P is a type family over A and that $p : x =_A y$. Then there is a lift of p, a function*

$$\mathsf{tr}^P(p) : P(x) \to P(y),$$

which ***transports from*** *$P(x)$* ***to*** *$P(y)$* ***over*** *p.*

Proof. By path induction it suffices to consider the case where y is x and p is refl_x, but in this case the identity function on $P(x)$ fits. □

Now we may formulate the action of dependent functions on paths.

Theorem 4.10 (Dependent action on paths)
Given a family of types P over A, then we have a ***dependent action on paths***

$$\mathsf{dap}^P : \prod_{(f:\prod_{x:A} P(x))} \prod_{(x,y:A)} \prod_{(p:x=_A y)} \left(\mathsf{tr}^P(p, f(x)) =_{P(y)} f(y)\right)$$

Proof. Again, by path induction it suffices to consider the case that y is x and p is refl_x, but in this case we have $P(x) \equiv P(y)$ and $\mathsf{tr}(p, f(x)) \equiv f(x)$. Hence $\mathsf{refl}_{f(x)}$ is a term of the desired type. □

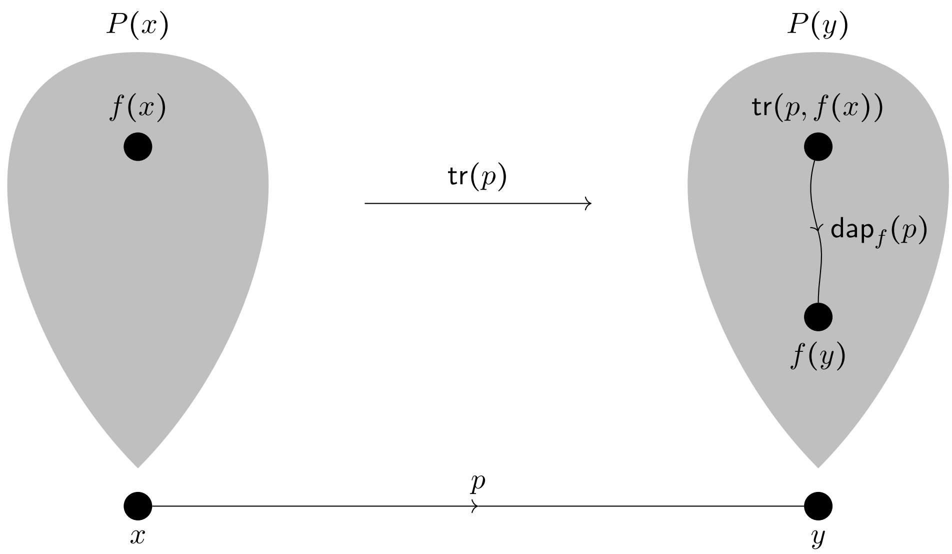


Figure 8: Homotopical interpretation of transport and dependent action on paths.

5 Higher inductive types

Higher inductive types are constructions that exploit the identity type to construct spaces with non-trivial homotopical data inside of Martin–Löf type theory.

In addition to having point generators, i.e. term introduction rules, higher inductive types also have path generators, i.e. introduction rules in their identity types. This poses a few complications in terms of their type rules. Identity types have predefined elimination and computation rules, so new constructors on them must be handled by the elimination and computation rules on the higher inductive type itself. But the elimination principle on a type is a way to construct functions from that type to another. Thus only through (potentially iterated) use of the function action on paths may one recover the higher path structure. Yet the function action on paths is also predefined through identity elimination, so we cannot postulate syntactic computation rules on it. We can however add new *propositional* computation rules, by giving witnesses to them. This is exactly what we do.

In this section, we survey a few aspects of higher inductive types, defining an interval and circle type. We use the first to demonstrate the consequences such definitions have on the type theory itself, and the second to demonstrate homotopy-theoretic aspects of the type theory.

These sections are intended only as a short survey of the concepts, and lacks rigor, frequently skipping arguments entirely. They do however visit many important concepts including function extensionality, equivalence, contractibility, and n-truncatedness, which will all make later reappearances.

5.1 The interval type

The interval $\mathtt{I}$ is a higher inductive type with the following introduction rules

$$0_{\mathtt{I}} : \mathtt{I} \qquad 1_{\mathtt{I}} : \mathtt{I} \qquad \mathsf{seg} : 0_{\mathtt{I}} =_{\mathtt{I}} 1_{\mathtt{I}} \tag{$\mathtt{I}$-introduction}$$

Using a diagramatical rendition, the structure of the interval immediately becomes clear.

$$0_{\mathtt{I}} \xrightarrow{\quad\mathsf{seg}\quad} 1_{\mathtt{I}}$$

Its recursion principle is the following: given a type A such that we have terms $a, a' : A$ with a path between them $p : a =_A a'$, then we have a function from the interval to A with the following computational rules:

$$\mathsf{rec}_{\mathtt{I}}(a, a', p, 0_{\mathtt{I}}) :\equiv a \qquad \mathsf{rec}_{\mathtt{I}}(a, a', p, 1_{\mathtt{I}}) :\equiv a' \qquad \mathsf{ap}_{\mathsf{rec}_{\mathtt{I}}(a,a',p)}(\mathsf{seg}) := p.$$

Notice that the last defining equality is only propositional, meaning that we only emit a homotopy

$$\eta : \mathsf{ap}_{\mathsf{rec}_{\mathtt{I}}(a,a',p)}(\mathsf{seg}) = p.$$

Remark 5.1. The interval covariantly represents the identity type

$$(\mathtt{I} \to A) \simeq \left(\sum_{(x,y:A)} x =_A y \right).$$

We can see this from the fact that a map from the interval to a type A consists of three pieces of data:

$$h : \quad 0_{\mathtt{I}} \mapsto a_0 \qquad 1_{\mathtt{I}} \mapsto a_1 \qquad \text{seg} \mapsto p \text{ (propositionally and by } \mathsf{ap}_h \text{)}$$

The interval is also contractible, by which we mean

Definition 5.2 (Contractibility)**.** We say a type X is **contractible** if the following type is inhabited

$$\mathsf{isContr}\, X :\equiv \sum_{(x:X)} \prod_{(y:X)} (x =_X y).$$

Which may be intuitively interpreted as proof that "there exists a point x such that for all points y there is a path to x." And we say that x is a center of contraction of X. Note that this choice of paths must depend continuously on x by construction.

Another canonical example of a contractible space, given an inhabited type $x : X$, is the **based path space**

$$\sum_{y:X} (x =_X y).$$

5.2 Function extensionality

One of the most notable features of intuitionistic type theory as opposed to intuitionistic set theory is that it makes a distinction between judgmental and propositional equality. The first can be interpreted as a consequence of our definitions. Precisely, it is the least equivalence preserved by our computation and uniqueness rules. Because of this, it is an *intensional* notion of equality: we give a collection of meanings to a term by how we define it, and another term is only judgmentally equal to this term if it has the same definitional meanings. This stands in opposition to the notion of *extensional* equality, in which two terms are considered equal if they relate to everything else in the same way.

Let us look at an example. Assume we have defined the following two functions on $\mathbb{N}$:

$$f(x) :\equiv 2x \quad \text{and} \quad g(x) :\equiv x + x$$

They are extensionally equal: they evaluate to the same value for every natural number. However, their intensional meaning is different: the first function takes the value x and multiplies it by two. The second function takes the value x and adds x to it. They have different algorithms, which is part of their intensional meaning.

It may be that it is the intensional meaning we are after, but usually, it is extensional equality that is desirable. Usually what we mean by a function is what values it takes, and not how it calculates them. In our current definition of the type theory, this is not the case. Equality is not extensional on functions. However, this may be expressed as an axiom.

To define this axiom, we first require a definition of equivalence of types.

Definition 5.3 (Function homotopy)**.** We say that two dependent functions which are equal on evaluations are **homotopic**, denoted as the type

$$f \sim g :\equiv \prod_{(x:A)} (f(x) =_{B(x)} g(x)).$$

Definition 5.4 (Equivalence)**.** Given two types A and B, we say that they are *equivalent* if the following type of *homotopy bi-invertible* maps is inhabited

$$A \simeq B :\equiv \sum_{(f:A\to B)} \mathsf{isEquiv}\, f \equiv \sum_{(f:A\to B)} \left(\sum_{(g:B\to A)} f \circ g \sim \mathrm{id}_B \right) \times \left(\sum_{(h:B\to A)} h \circ f \sim \mathrm{id}_A \right).$$

Warning 5.5 (Homotopy coherent equivalences)**.** The critical reader may question the use of left- and right-inverses instead of a two-sided inverse in this definition. The reason is that with our definition we receive a better-behaved equivalence type. In particular, a proof of bi-invertibility contains no more data than the mere truth that it is bi-invertible: the type of proofs is (-1)-truncated.[9] Had we instead defined equivalence using mutual inverses we would have gotten the notion of *quasi-inverses.* There is a subtle issue with this definition. The type of quasi-inverses may have higher non-trivial

[9] We give a precise definition of this later.

data [UF13, Lemma 4.1.1 and Theorem 4.1.3], and if we formulate other axioms in terms of quasi-inverses we could even make the type theory inconsistent. This is for instance the case for univalence.

Having defined a homotopy coherent notion of equivalence, we are ready to state the function extensionality axiom:

Axiom 4 (Function extensionality)**.** *Given a type family* $x : A \vdash B\ \mathsf{type}$ *and maps* $f, g : \prod_{x:A} B(x)$*, the canonical map*

$$\mathsf{id\text{-}to\text{-}htp}(f,g) : (f = g) \to (f \sim g).$$

defined by path induction $\mathsf{id\text{-}to\text{-}htp}(f,f,\mathsf{refl}_f) :\equiv (x \mapsto \mathsf{refl}_{f(x)})$ *is an equivalence with witness*

$$\mathsf{funext} : \prod_{f,g:\prod_{x:A} B(x)} \mathsf{isEquiv}(\mathsf{id\text{-}to\text{-}htp}(f,g)).$$

Hence we have the reading: "equality on functions is equality on evaluations."

To be precise, what we mean by *assuming* such an axiom is to adjoin the following inference rule to the type theory:

$$\frac{\Gamma \vdash A\ \mathsf{type} \qquad \Gamma, x : A \vdash B\ \mathsf{type}}{\Gamma \vdash \mathsf{funext}_{(\prod_{x:A} B)} : \prod_{f,g:\prod_{x:A} B(x)} \mathsf{isEquiv}(\mathsf{id\text{-}to\text{-}htp}(f,g))}(\mathsf{funext}).$$

In the presence of an interval type, we can in fact *prove* function extensionality, as the interval type covariantly represents path types and hence gives us an object to curry over.

Theorem 5.6
The existence of the interval type implies function extensionality.

Proof. We prove the existence of a map $(f \sim g) \to (f = g)$ and leave the proof that it defines an inverse to $\mathsf{id\text{-}to\text{-}htp}(f,g)$ as an exercise. Hence we need to construct a function

$$\left(\prod_{x:A} f(x) =_{B(x)} g(x)\right) \to (f =_{(\prod_{x:A} B(x))} g)$$

So let us suppose we have a term $h : \left(\prod_{(x:A)} f(x) =_{B(x)} g(x)\right)$. By elimination on the interval we may define the following function

$$H :\equiv (t, x \mapsto \mathsf{rec}_{\mathtt{I}}(f(x), g(x), h(x), t)) : \mathtt{I} \to \prod_{x:A} B(x).$$

Notice that we have immediately permuted the arguments x and t. Evaluating H at zero we compute

$$H(0_{\mathtt{I}}) \equiv (x \mapsto \mathsf{rec}_{\mathtt{I}}(f(x), g(x), h(x), 0_{\mathtt{I}})) \overset{\mathtt{I}\text{-comp}}{\equiv} (x \mapsto f(x)) \overset{\Pi\text{-uniq}}{\equiv} f$$

and similarly $H(1_{\mathtt{I}}) \equiv g$. But then $\mathsf{ap}_H(\mathsf{seg}) : f = g$ is our desired witness. □

It may be noted that much of the theory can be developed in the absence of function extensionality, instead falling back to the weaker notion of function homotopy. This relation is still a **congruence**, meaning it is a substitutive equivalence relation.

5.3 The circle type

To get a small taste of the synthetic homotopy theory inside homotopy type theory we take a brief look at a type representing the homotopical circle.

The circle type $\mathtt{S}^1$ has the following introduction rules

$$\mathsf{base} : \mathtt{S}^1 \qquad \mathsf{loop} : \mathsf{base} =_{\mathtt{S}^1} \mathsf{base}$$

Its recursion principle states that to construct a function from the circle, one must give a term x and an identity term $p : x =_A x$.

Hence the recursion principle can be seen as stating that $\mathtt{S}^1$ is freely generated by a base point and a self-loop. It is this free generation that makes the homotopy structure of the circle type non-trivial.

Definition 5.7 (Loop space)**.** Given a pointed type $x : X$, i.e. a type with a distinguished point, we define the **loop space** to be

$$\Omega X :\equiv (x =_X x).$$

In particular, the loop space itself is canonically pointed with base-point the reflexivity.

It can be shown in homotopy type theory that the loop space of $\mathtt{S}^1$ is the integers $\mathbb{Z}$. For instance, this can be done using univalence and the *encode-decode method*, a new method for computing loop spaces and homotopy groups inside of homotopy type theory.

Theorem 5.8 ([UF13, Corollary 8.1.10])
The loop space of $\mathtt{S}^1$ *is equivalent to* $\mathbb{Z}$*.*

This tells us that this type has a higher non-trivial homotopical structure, a property which more generally can be formalized using the following notion:

Definition 5.9 (n-truncatedness)**.** We say a type X is **n-truncated**, or that it is an **n-type**, if the inductively defined type $\mathsf{is}\text{-}n\text{-}\mathsf{type}\, X$ is inhabited:

$$\mathsf{is}\text{-}(n+1)\text{-}\mathsf{type}\, X :\equiv \prod_{x,y:X} \mathsf{is}\text{-}n\text{-}\mathsf{type}(x =_X y) \qquad \mathsf{is}\text{-}(-2)\text{-}\mathsf{type}\, X :\equiv \mathsf{isContr}\, X.$$

In particular we call (-1)-types **propositions** and 0-types **sets**.

Corollary 5.10. *The circle* $\mathtt{S}^1$ *is* 1*-truncated.* □

Many homotopy-theoretic results have been proven in this synthetic homotopy theory. For instance, using higher inductive procedures we may define homotopy colimits like homotopy coequalizers, homotopy pushouts, and as a special case suspension. In particular, we can show $\pi_{k<n}(\mathtt{S}^n) \simeq \mathbf{1}$ and the existence of the Hopf fibration. Other results that have been formalized include the long-exact sequence of homotopy groups, the Van Kampen theorem, and the Blakers-Massey theorem. Whitehead's theorem does not hold, however, which can be traced back to the fact that not all $(\infty, 1)$-toposes are *hypercomplete*.

6 The universe type and univalence

In this section, we axiomatize a universe type and present the univalence axiom due to Vladimir Voevodsky. The univalence axiom captures the mathematician's intuition that equivalent things behave the same. Using the identity types of the universe and the fact that all constructions are homotopy invariant, this idea is very efficiently encoded in the type theory. As consequences, we look at the interpretation of functions as fibrations and the universal property of the universe type as an object classifier of small types.

To begin, we axiomatize a universe type in the style of Tarski, assuming closure under all of the basic type formers. The types viewed as terms of the universe are really representative codes for their corresponding type. For these codes, we temporarily introduce a quotation mark notation. After the axiomatization, we immediately forget this notation and treat terms of $\mathcal{U}$ directly as types.

$$\frac{\Gamma\ \mathsf{ctx}}{\Gamma \vdash \mathcal{U}\ \mathsf{type}}(\mathcal{U}\text{-form}) \qquad \frac{\Gamma \vdash A : \mathcal{U}}{\Gamma \vdash \mathsf{t}(A)\ \mathsf{type}}(\mathcal{U}\text{-elim}) \qquad \frac{\Gamma \vdash \text{“}A\text{”} : \mathcal{U}}{\Gamma \vdash \mathsf{t}(\text{“}A\text{”}) \equiv A\ \mathsf{type}}(\mathcal{U}\text{-uniq})$$

$$\frac{\Gamma \vdash A : \mathcal{U} \qquad \Gamma, x : A \vdash B : \mathcal{U}}{\Gamma \vdash \text{“}\textstyle\prod_{x:A} B\text{”} : \mathcal{U} \qquad \Gamma \vdash \text{“}\textstyle\sum_{x:A} B\text{”} : \mathcal{U}} \qquad \frac{\Gamma \vdash A : \mathcal{U} \qquad \Gamma \vdash a_1 : A \qquad \Gamma \vdash a_2 : A}{\Gamma \vdash \text{“}a_1 =_A a_2\text{”} : \mathcal{U}}$$

$$\frac{\Gamma\ \mathsf{ctx}}{\Gamma \vdash \text{“}\mathbf{0}\text{”} : \mathcal{U} \qquad \Gamma \vdash \text{“}\mathbf{1}\text{”} : \mathcal{U} \qquad \Gamma \vdash \text{“}\mathbf{2}\text{”} : \mathcal{U}}$$

Figure 9: Rules for the universe type

We note the assumption of closure under the formation of the empty, unit and two-term type. Equivalently, we could assume closure under binary products, binary sums and the formation of only the empty and unit type. In a richer type theory one would naturally include closure under the formation of W-types and various colimits such as higher inductive types (closure under limits follows from the current closure properties). However, as we have not defined these in any generality we are left with the current definition.

6.1 Univalence

When we internalize a universe types, all of the type formers apply as with any other type, so $\mathcal{U}$ itself also becomes an ∞-groupoid. In particular we may define the following function by path induction:

$$\mathsf{id\text{-}to\text{-}eqv} : (A =_{\mathcal{U}} B) \to (A \simeq B) \qquad \mathsf{id\text{-}to\text{-}eqv}(\mathsf{refl}_A) :\equiv \mathrm{id}_A\,.$$

This directly leads us to the univalence axiom:

Axiom 5 (Univalence axiom)**.** *The function* $\mathsf{id\text{-}to\text{-}eqv}$ *is an equivalence*

$$(A =_{\mathcal{U}} B) \simeq (A \simeq B).$$

"Equivalent types may be identified."

Decomposing a general equivalence, we may separate the univalence axiom into four principles of natural deduction:

ua**-introduction:** An introduction rule for identity terms, $\mathsf{ua} : (A \simeq B) \to (A =_{\mathcal{U}} B)$.

ua**-elimination:** An elimination rule on identity terms, $\mathsf{id\text{-}to\text{-}eqv}$.

ua**-computation:** The propositonal computation rule $\mathsf{tr}(\mathsf{ua}(f), x) = f(x)$.

ua**-uniqueness:** The propositional uniqueness rule $p = \mathsf{ua}(\mathsf{tr}(p))$.

Though of course, the elimination principle holds regardless of the assumption of the axiom.

Mathematical efficiency. Univalence can be seen as an axiom of mathematical efficiency. By use of the equality type, when proving a theorem for a particular type, it allows us to automatically prove that same theorem for all equivalent types by transporting along the corresponding equality term, instead of having to write individual proofs of transport along particular equivalences.

Classifying space. As with all constructions in homotopy type theory, we would expect univalence to have an interpretation in homotopy theory. Indeed, in this sense, univalence expresses the universe as the classifying space of the disjoint sum of small homotopy types.

Univalent categories. Another interpretation is that univalence encodes the idea that sometimes isomorphism and not just equality is the "right" notion of identification. This aspect of univalence inspires the theory of *precategories*, which are categories with an open-ended system of identification. We may impose a univalence condition on them, identifying isomorphic objects to obtain *categories* in the terminology of [UF13]. In the case of univalent categories, one for instance has that equivalences of categories are exactly the functors which are fully faithful and essentially surjective, without depending on any formulation of the axiom of choice. The additional liberty that comes with a theory of *precategories* is that it leaves this question of identification open to the mathematician.

Let us briefly showcase some elementary consequences of the univalence axiom.

Example 6.1. There is a non-trivial equivalence on $\mathbf{2}$, the two-term type. Define the map swap as follows using $\mathbf{2}$-recursion:

$$\mathsf{swap} : \mathbf{2} \to \mathbf{2} \quad \mathsf{swap}(0) :\equiv 1 \quad \mathsf{swap}(1) :\equiv 0$$

Then $\mathsf{swap}(\mathsf{swap}(x)) \equiv x$, so swap is an equivalence of types which is its own inverse.

Corollary 6.2. *The identity type* $\mathbf{2} =_{\mathcal{U}} \mathbf{2}$ *is not contractible.* $\square$

Corollary 6.3. *The universe type* $\mathcal{U}$ *is not* 0*-truncated.* $\square$

We can also use this example to show the following.

Theorem 6.4 ([UF13, Corollary 3.2.7])
Univalence is inconsistent with the naive interpretation of the law of excluded middle.

Heuristic argument. Interpreting the law of excluded middle directly using the propositions as types correspondence we get

$$\mathsf{lem} : \prod_{X:\mathcal{U}} X + (X \to \mathbf{0}).$$

This means that we have a function picking out an inhabitant of any non-empty type. In particular, it must pick out a term $\mathsf{lem}(\mathbf{2}) : \mathbf{2}$. Now, by univalence, such a function must be invariant under equivalences. So in particular, the law of excluded middle must be invariant under the permutation, swap. However, it may be proven that $0 \neq 1$, and so no matter which term is chosen by lem we have a contradiction. $\square$

To circumvent this incompatibility, we may formulate a restricted law of excluded middle which more closely resembles the classical formulation (refering back to Digression 1.4)

$$\mathsf{lem}_{-1} : \prod_{(X,\rho):\mathsf{Prop}\,\mathcal{U}} X + (X \to \mathbf{0}),$$

where $\mathsf{Prop}\,\mathcal{U}$ denotes the type of propositions:

$$\mathsf{Prop}\,\mathcal{U} :\equiv \sum_{X:\mathcal{U}} \mathsf{is}\text{-}(-1)\text{-}\mathsf{type}\,X.$$

In particular, in the context of the previous argument, we see that this formulation ensures that whichever term lem_{-1} picks out of an inhabited type is equal to every other term of that type. Hence our previous argument no longer applies.

This formulation of the law of excluded middle is consistent with univalence, and it may be noted that many classical arguments depending on the law of excluded apply with this formulation. Similarly, we may formulate a version of the classical axiom of choice using a propositional truncation operation which is also consistent with homotopy type theory.

Another consequence of the univalence axiom is that, like with the existence of an interval type, we get function extensionality for free.

Theorem 6.5 ([UF13, Theorem 4.9.4])
Univalence implies function extensionality.

6.2 Functions are fibrations

In the presence of an internal universe, we may also internalize type families $x : B \vdash P : \mathcal{U}$ as function terms $P : B \to \mathcal{U}$. These satisfy the homotopy lifting property:

Proposition 6.6 (Homotopy lifting property)**.** *Given a type family $P : B \to \mathcal{U}$, we have a path lifting map*

$$\mathsf{lift} : \prod_{(x:B)} \prod_{(u:P(x))} \prod_{(y:B)} \prod_{(p:x=_B y)} \big((x,u) =_{(\Sigma_{x:B} P(x))} (y, \mathsf{tr}(p,u))\big).$$

Proof. By path induction it suffices to consider the case where y is x and p is refl_x, in which case $\mathsf{tr}(p,u)$ computes to u and so we have the canonical equality witness $\mathsf{refl}_{(x,u)}$. □

Hence a type family $P : B \to \mathcal{U}$ contains the data of a fibration with base space B and total space $\sum_{x:B} P(x)$. The fiber over $b : B$ is simply $P(b)$. This property may be extended to all maps of small types via the homotopy fiber construction. Hence as we will shortly display, maps of small types are fibrations.

Definition 6.7 (Homotopy fiber)**.** Given a map of small types $\pi : E \to B$ we may define the **homotopy fiber family** as

$$\mathsf{fib}_\pi :\equiv b \mapsto \left(\textstyle\sum_{e:E} (\pi(e) =_B b)\right) : B \to \mathcal{U}.$$

In the presence of a univalent universe, this construction is part of an equivalence between type families and functions, using mutual inverse constructions called straightening and unstraightening.

Definition 6.8 (Fibrations and families)**.** We define the type of fibrations and type of type families in $\mathcal{U}$ as

$$\mathsf{Fib}\,\mathcal{U} :\equiv \sum_{B,E:\mathcal{U}} (E \to B) \qquad \text{and} \qquad \mathsf{Fam}\,\mathcal{U} :\equiv \sum_{B:\mathcal{U}} (B \to \mathcal{U}).$$

Hence we define a fibration to be any function between small types.

Theorem 6.9 (Straightening and unstraightening, [UF13, Theorem 4.8.3])
Given a univalent universe $\mathcal{U}$ and a small type B we have mutual inverse maps st *and* un *between fibrations over B and families over B:*

$$\mathsf{st}\,\pi :\equiv (B, \mathsf{fib}_\pi) \qquad \textit{and} \qquad \mathsf{un}\,P :\equiv (B, \textstyle\sum_{b:B} P(b), \mathsf{pr}_1)$$

$$\begin{tikzcd}
\mathsf{Fib}\,\mathcal{U} \arrow[rr, bend left, "\mathsf{st}"] \arrow[rr, phantom, "\simeq"] \arrow[dr, "\mathsf{pr}_1"'] & & \mathsf{Fam}\,\mathcal{U} \arrow[ll, bend left, "\mathsf{un}"] \arrow[dl, "\mathsf{pr}_1"] \\
& \mathcal{U}. &
\end{tikzcd}$$

As a consequence, a univalent universe type satisfies the universal property of being the object classifier of small types.

Corollary 6.10 (Object classifier [UF13, Theorem 4.8.4]). *The univalent universe type is the object classifier of small types with universal fibration*

$$\mathsf{pr}_1 : \sum_{X:\mathcal{U}} X \to \mathcal{U}.$$

By this we mean that for any fibration π or type family P, which correspond to one another under the straightening-unstraightening construction, the following square is a pullback[10]

$$\begin{tikzcd} \sum_{b:B} P(b) \arrow[r, "(b,p)\mapsto(P(b),p)"] \arrow[d, "\pi"'] \arrow[dr, phantom, "\lrcorner", very near start] & \sum_{X:\mathcal{U}} X \arrow[d, "\mathsf{pr}_1"] \\ B \arrow[r, "P"] & \mathcal{U}. \end{tikzcd}$$

[10] We will give a precise definition of pullbacks in homotopy type theory in Theorem 15.22.

Part II
Simplicial Homotopy Theory

In this part, we explore a multitude of facets of objects called simplicial sets. They form a unifying framework inside of which we may discuss the theory of both categories and homotopy spaces in full generality. To better understand these objects and the connections between ideas, we spend much time developing the language of model categories.

We first introduce the simplex category $\mathbf{\Delta}$, which is the category of finite ordinals and order-preserving maps. Geometrically, its objects can be understood as directed simplexes and its maps as face inclusions and degeneration maps.

We then move to the study of **Set**-valued presheaves on $\mathbf{\Delta}$, called simplicial sets. Extending this geometric interpretation to all simplicial sets, they provide combinatorial descriptions of CW-complexes. This interpretation is concretized by the geometric realization functor. Conversely, topological spaces can be interpreted as simplicial spaces by means of the singular simplicial complex construction.[11] Simplicial sets constructed in this way satisfy a certain condition we call the Kan condition. It is a remarkable fact, first proven by Quillen [Qui67], that this condition is part of a model of homotopy types equivalent to that exhibited by point-set topology. This model forms the motivating basis for our discussion of many topics in this part, hence we will frequently revisit it in the context of new ideas.

We can also interpret categories in simplicial sets by means of the nerve construction. As simplicial sets, they satisfy a condition we call the Segal condition.

After having discussed these fundamental ideas in simplicial sets, we move on to discuss what we in general terms may consider a *homotopy theory*. To this end, we introduce the very general concept of homotopical categories taking inspiration from [Rie20]. Inside these, we may formulate surprisingly many concepts familiar from homotopy theory, including the homotopy category, derived functors, and homotopy limits.

Homotopical categories in their full generality exhibit very little structure, and so are in general unwieldy to work with. When working with richly structured categories such as the category of topological spaces or the category of simplicial sets, a more powerful machinery is required. To this end, we introduce the incredibly useful language of model categories as first introduced by Quillen [Qui67]. These present a well-structured homotopy theory of the underlying homotopical category by means of classes of fibrations and cofibrations. To define these, we use the abstraction of weak factorization systems which we turn our attention to first.

After this, we are ready to define and discuss model categories and their model structures. In particular, we discuss notions of morphism homotopy which form the basis for an alternative construction of the homotopy category. In the course of this section, we visit a series of properties one may have on a model category, including cofibrant generation, enrichment, reedy model structures, projectivity and injectivity of model

[11] In fact, these constructions form an appropriate sort of equivalence between the homotopy theories of the categories.

structures, and properness.

Finally, we discuss two applications of the language of model categories and the Kan–Quillen model structure on simplicial sets. One application is in giving semantics to homotopy type theory, solidifying our understanding of types as homotopy types and giving classical homotopy theoretic meaning to theorems in homotopy type theory. The other application is in formulating the model of Rezk spaces, also known as complete Segal spaces, for the homotopy theory of $(\infty, 1)$-categories. These give semantics to the type theory we consider in the next part.

There is much more that could be said about everything we discuss in this part, and we simply cannot cover it all, or even as much as we would like. For the sake of expedience, we frequently skip the more involved or repetitive arguments, instead referring the reader to an external source.

If the reader is already familiar with the theory of model categories, the Kan–Quillen model structure on simplicial sets, the Reedy model structure, and the nerve construction, they may safely skip to Section 11, in which we discuss models of type theory.

7 Simplicial sets

In this section, we define the simplex category, simplicial sets and discuss various constructions on these including the nerve construction, geometric realization, and the singular simplicial complex. The contents are mostly based on [Fri08] and [Ras18].

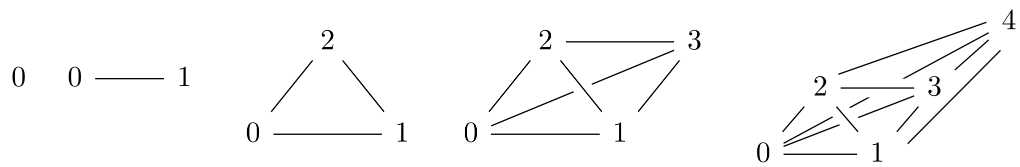
7.1 Directed simplices

For every $n \in \mathbb{N}$ (including zero) we define the category $[n]$ as the total order category given by the finite ordinal of order n

$$\{0 \leq 1 \leq \cdots \leq n\},$$

meaning that we have the objects $0, \ldots, n$ and we have precisely one arrow $x \to y$ in $[n]$ if $x \leq y$, and otherwise none. These categories constitute the objects of the **simplex category $\mathbf{\Delta}$** whose morphisms are order preserving maps. Equivalently, $\mathbf{\Delta}$ may be regarded as the full subcategory of **Cat**, the category of small categories, consisting of these objects.

In $\mathbf{\Delta}$, the object $[n]$ may be interpreted as a filled n-dimensional directed simplex. Here are the first few objects depicted as such. Note that every edge is and points from the smaller to the larger of its vertices.



We have two elementary classes of maps in this category:

Definition 7.1 (Elementary face and degeneracy maps)**.** Let n and k be natural numbers such that $k \leq n$. Then we define the following $\mathbf{\Delta}$-morphisms

$$d_k x := \begin{cases} x & x < k \\ x + 1 & x \geq k \end{cases} \quad : \ [n-1] \to [n] \qquad \text{(Elementary face maps)}$$

$$s_k x := \begin{cases} x & x \leq k \\ x - 1 & x > k \end{cases} \quad : \ [n+1] \to [n] \qquad \text{(Elementary degeneracy maps)}$$

These maps again have a concrete spatial interpretation. The elementary face map d_k includes a simplex-face opposite the k'th vertex in its codomain while the elementary degeneracy map s_k collapses the face opposite the k'th vertex.

Lemma 7.2 (Simplicial relations)**.** *Elementary face and degeneracy maps satisfy the following relations*

$$d_i d_j = d_{j+1} d_i \ \textit{if } i \leq j \qquad s_i s_j = s_j s_{i+1} \ \textit{if } j \leq i \qquad d_i s_j = \begin{cases} s_{j-1} d_i & i < j \\ \mathrm{id} & i = j \vee i = j+1 \\ s_j d_{i-1} & i > j+1 \end{cases} .$$

Proof. This may be verified by direct computation. □

Lemma 7.3 (Epi-mono factorization)**.** *Every morphism in $\mathbf{\Delta}$ may be factored as a finite composition of elementary degeneracy maps followed by a finite composition of elementary face maps*

$$f = d_{i_u} \dots d_{i_1} s_{j_v} \dots s_{j_1} \qquad \textit{such that for } k < k' \textit{ we have } i_k < i_{k'} \textit{ and } j_k \geq j_{k'}.$$

Proof. Let f be a morphism from $[n]$ to $[m]$. The map f may be identified with a monotone sequence of natural numbers less than m of length n,

$$0 \leq f(0) \leq f(1) \leq \cdots \leq f(n-1) \leq f(n) \leq m.$$

Let $i_1, \dots, i_u$ be the numbers which are not in this sequence in rising order, and let $j_1, \dots, j_v$ be the indices at which f does not increase, $f(j) = f(j+1)$ in decreasing order, then we may verify the result by direct computation. □

Corollary 7.4. *The elementary face maps form a generating set for the monomorphisms of $\mathbf{\Delta}$, and the elementary degeneracy maps form a generating set for the epimorphisms.*

We may also interchange the roles of face and degeneracy maps of the previous lemma.

Corollary 7.5 (Mono-epi factorization)**.** *Every morphism in $\mathbf{\Delta}$ may be factored as a finite composition of elementary face maps followed by a finite composition of elementary degeneracy maps.*

Proof. Apply the previous lemma and then interchange face and degeneracy maps using the simplicial relations in an inductive procedure. □

As another particular consequence, the categorical structure on $\mathbf{\Delta}$ is completely determined by the elementary face and degeneracy maps. Hence in categorical considerations, it suffices to examine what happens at the elementary face and degeneracy maps. This manifests itself in the following common diagrammatic depiction of $\mathbf{\Delta}$, where right-pointing arrows are elementary face maps and left-pointing arrows are elementary degeneracy maps:

$$\begin{tikzcd} {[0]} \arrow[r, shift left=2, "d_0"] \arrow[r, shift right=2, "d_1"'] & {[1]} \arrow[l, "s_0" description] \arrow[r, shift left=4] \arrow[r] \arrow[r, shift right=4] & {[2]} \arrow[l, shift left=2] \arrow[l, shift right=2] \arrow[r, shift left=6] \arrow[r, shift left=2] \arrow[r, shift right=2] \arrow[r, shift right=6] & \cdots \arrow[l, shift left=4] \arrow[l] \arrow[l, shift right=4] \end{tikzcd}$$

Construction 7.6 (Geometric simplex realization)**.** We have a functor $|{-}|$ which takes n-dimensional directed simplices and embeds them as topological n-dimensional simplices in $\mathbb{R}^n$

$$|{-}| : \mathbf{\Delta} \to \mathbf{Top}, \qquad [n] \mapsto \{(x_1, \ldots, x_n) \in \mathbb{R}^n \mid 0 \le x_n \le \cdots \le x_1 \le 1\}.$$

7.2 Simplicial sets

Having familiarized ourselves with the simplex category, we now move on to study **Set**-valued presheaves on it, **simplicial sets**. While a seemingly innocuous idea, these objects are incredibly versatile. Particularly important to us, they may be used to define models of ∞-groupoids and $(\infty, 1)$-categories. We begin by discussing the basics and giving some fundamental examples.

Definition 7.7 (Simplicial Set)**.** A **simplicial set** is a functor

$$\mathbf{\Delta}^{\mathrm{op}} \to \mathbf{Set}.$$

They form the objects of the category of simplicial sets **sSet** whose morphisms are natural transformations. We will denote $X([n])$ as X_n for X a simplicial set.

As with $\mathbf{\Delta}$, we may think of X in terms of what happens under the face and degeneracy maps, which are now reversed.

$$\begin{tikzcd} X_0 \arrow[r, "s_0" description] & X_1 \arrow[l, shift right=2, "d_0"'] \arrow[l, shift left=2, "d_1"] \arrow[r, shift left=2] \arrow[r, shift right=2] & X_2 \arrow[l, shift left=4] \arrow[l] \arrow[l, shift right=4] \arrow[r, shift left=4] \arrow[r] \arrow[r, shift right=4] & \cdots \arrow[l, shift left=6] \arrow[l, shift left=2] \arrow[l, shift right=2] \arrow[l, shift right=6] \end{tikzcd}$$

We call X_0 the **vertices** or **0-simplices** of X, and X_1 the **directed edges** or **arrows** or **1-simplices** of X. In generality we call X_n the n**-simplices** of X.

Example 7.8 (Represented[12] simplicial sets)**.** The Yoneda embedding of $\mathbf{\Delta}$ into **sSet** constitutes an important class of simplicial sets called the **standard** n**-simplexes**, denoted Δ^n

$$\Delta^n := \mathrm{Hom}_{\mathbf{\Delta}}(-, [n]).$$

As a bonus, by the Yoneda lemma, we get bijections $\mathrm{Hom}_{\mathbf{sSet}}(\Delta^n, X) \cong X_n$ natural in both $[n]$ and X.

[12]We use this wording as opposed to *representable* to emphasize the constructive aspect: the representing object is specified.

Other central examples of simplicial sets are the boundary, horns, and spine of the standard simplices:

Example 7.9 (Boundary, horns, and spine)**.** Given a natural number n we may define the following simplicial subsets of the standard n-simplex Δ^n:

$\partial\Delta^n$ The **boundary** $\partial\Delta^n$ is the simplicial subset only missing the nondegenerate n-cell of Δ^n.

Λ^n_k Given a natural number k such that $0 \le k \le n$, then the **k'th horn Λ^n_k of** Δ^n, is the simplicial subset of Δ^n consisting of all but the k'th $(n-1)$-dimensional faces of Δ^n.

Sp^n The **spine** Sp^n **of** Δ^n is the simplicial subset consisting of only the directed edges between consecutive vertices $0 \to 1, 1 \to 2$ up to $(n-1) \to n$.

In particular we have the inclusions

$$\mathrm{Sp}^n \overset{(*)}{\hookrightarrow} \Lambda^n_k \hookrightarrow \partial\Delta^n \hookrightarrow \Delta^n$$

where $\mathrm{Sp}^n \hookrightarrow \Lambda^n_k$ exists when $n > 2$ or $n = 2$ and $k = 1$ in which case we may note that $\mathrm{Sp}^2 = \Lambda^2_1$. We can draw the 3-horn of Δ^3 as follows, imagining its faces to be opaque:

$$\begin{tikzcd}
& 2 \arrow[rr, dash] \arrow[dl, dash] \arrow[dr, dash] & & 3 \arrow[dl, dash] \\
0 \arrow[rr, dash] \arrow[urrr, dash] & & 1 .
\end{tikzcd}$$

It has the same vertices and edges as Δ^3, but the front face is removed and the interior is hollow. The spine of Δ^3 on the other hand is the following simplicial subset

$$\begin{tikzcd}
& 2 \arrow[rr, dash] \arrow[dr, dash] & & 3 \\
0 \arrow[rr, dash] & & 1 .
\end{tikzcd}$$

The interpretation of simplicial sets as spaces is easily substantiated through their interpretation as combinatorial descriptions of CW-complexes. This is what the geometric realization functor does for us:

Construction 7.10 (Geometric realization and singular complex)**.** We have a left Kan extension of the geometric simplex realization functor $|-| : \mathbf{\Delta}^{\mathrm{op}} \to \mathbf{Top}$ along the Yoneda embedding $\mathbf{\Delta}^{\mathrm{op}} \to \mathbf{sSet}$ which gives us a **geometric realization** for all simplicial sets

$$|-| : \mathbf{sSet} \to \mathbf{Top}.$$

This funtor takes a simplicial set and interprets it as a combinatorial description of a CW-complex:

$$|X| := \coprod_{n \in \mathbb{N}} (|[n]| \times X_n)/\sim$$

Where $|[n]|$ is the geometric realization of the directed n-simplex as in Construction 7.6 and $\sim$ is the equivalence relation generated by $(fp, x) \sim (p, fx)$ for all morphisms f in $\mathbf{\Delta}$. This construction has a right adjoint known as the singular simplicial complex functor:

$\mathrm{S}_\bullet : \mathbf{Top} \to \mathbf{sSet}$ defined levelwise as the sets of continuous functions $\mathrm{S}_n(X) := X^{|[n]|}$

and with the obvious face and degeneracy maps.

Proposition 7.11. *The category* **sSet** *is cartesian closed.*

Proof. This is just a special case of the fact that all **Set**-valued presheaves are cartesian closed. See for instance [RV21, Example A.1.4 (iii)]. □

And so in particular **sSet** is enriched over itself. In fact, since (co)limits in a presheaf category over a (co)complete category are determined degreewise, we have the following descriptions of the product and exponential:

Definition 7.12 (Products and internal homs in **sSet**)**.** Given simplicial sets X and Y, then their product and exponential may be defined degreewise to be

$$(X \times Y)_n = X_n \times Y_n \qquad \text{and} \qquad (Y^X)_n = \mathrm{Hom}_{\mathbf{sSet}}(\Delta^n \times X, Y).$$

7.3 The nerve construction

Categories may be interpreted as certain simplicial sets under the *nerve construction*. Under this construction, their objects are understood as vertices, and morphisms are understood as directed edges:

Definition 7.13 (Nerve construction $\mathcal{N}$)**.** Given a small category $\mathcal{C}$, the **nerve construction** is the composite presheaf

$$\mathrm{Hom}_{\mathbf{Cat}}(i(-), \mathcal{C}) \in \mathrm{Ob}(\mathbf{sSet}),$$

where i is the inclusion of the simplex category $\mathbf{\Delta}$ into **Cat**. This construction can be understood levelwise as

$$(\mathcal{N}\mathcal{C})_n = \mathrm{Hom}_{\mathbf{Cat}}([n], \mathcal{C})$$

i.e. $(\mathcal{N}\mathcal{C})_0$ are objects of $\mathcal{C}$, $(\mathcal{N}\mathcal{C})_1$ are arrows and in general $(\mathcal{N}\mathcal{C})_n$ are sequences of composable morphisms of length n in $\mathcal{C}$. Note that the size condition ensures that $\mathrm{Hom}_{\mathbf{Cat}}([n], \mathcal{C})$ is a (small) set for every n.

Now, by precomposition it is clear what the elementary face and degeneracy operations do: the face operation d_k composes a pair of morphisms at the k'th position while the degeneracy operation s_k inserts an identity morphism. Explicitly, given $0 \le k \le n$ we may describe them on arrows between consecutive vertices as follows

$$d_k^{\mathcal{N}\mathcal{C}}(f)(i \le i+1) = \begin{cases} f(i \le i+1) & i < k \\ f(i \le i+2) & i = k \\ f(i+1 \le i+2) & i > k \end{cases} \qquad : \mathcal{C}^{[n]} \to \mathcal{C}^{[n-1]}$$

$$s_k^{\mathcal{N}\mathcal{C}}(f)(i \le i+1) = \begin{cases} f(i \le i+1) & i < k \\ \mathrm{id} & i = k \\ f(i-1 \le i) & i > k \end{cases} \qquad : \mathcal{C}^{[n]} \to \mathcal{C}^{[n+1]}.$$

As a functorial construction, the nerve also has a defined action on functors $F : \mathcal{C} \to \mathcal{D}$. This simplicial map is understood as just levelwise postcomposition

$$\mathcal{N} F_n = F \circ - : \mathcal{C}^{[n]} \to \mathcal{D}^{[n]}.$$

A natural question is to ask when a simplicial set may be obtained as the nerve of a category. For any simplicial set X we have a restriction map

$$\mathrm{Hom}_{\mathbf{sSet}}(\Delta^n, X) \to \mathrm{Hom}_{\mathbf{sSet}}(\mathrm{Sp}^n, X)$$

Equivalently, this map may be constructed using the obvious bijection

$$X_1 \times_{X_0} \cdots \times_{X_0} X_1 \cong \mathrm{Hom}_{\mathbf{sSet}}(\mathrm{Sp}^n, X)$$

and the following cone

$$\begin{tikzcd}
 & & & & X_n \arrow[llll, bend right=10, "d_2^{n-1}"'] \arrow[dll, "d_2^{n-2}d_0"'] \arrow[d, dotted] \arrow[drr, "d_2d_0^{n-2}"] \arrow[rrrr, bend left=10, "d_0^{n-1}"] & & & & \\
X_1 \arrow[dr, "d_1"'] & & X_1 \arrow[dl, "d_0"] \arrow[dr, "d_1"'] & & \cdots \arrow[dl] \arrow[dr] & & X_1 \arrow[dl, "d_0"] \arrow[dr, "d_1"'] & & X_1 \arrow[dl, "d_0"] \\
 & X_0 & & X_0 & \cdots & X_0 & & X_0 &
\end{tikzcd}$$

noting that

$$d_2^{n-i-1} d_0^i (x_0 \xrightarrow{f_0} \dots \xrightarrow{f_{n-1}} x_n) = f_i.$$

Hence commutativity of the cone is witnessed by the fact that coinciding vertices of consecutive edges agree. For instance, in the case of $n = 2$ this is the cone in the below-left diagram whose commutativity expresses that for any 2-cell $\sigma \in X_2$ the middle vertex may be expressed as two composites of face maps as depicted below-right

$$\begin{tikzcd}
 & X_2 \arrow[dl, "d_0"'] \arrow[dr, "d_2"] & \\
X_1 \arrow[dr, "d_1"'] & & X_1 \arrow[dl, "d_0"] \\
 & X_0 &
\end{tikzcd}
\qquad
\begin{tikzcd}
 & d_0d_2\sigma = v_1 = d_1d_0\sigma \arrow[d, phantom, "\sigma"] & \\
v_0 \arrow[ur, dash, "d_2\sigma"] \arrow[rr, dash] & {} & v_2 \arrow[ul, dash, "d_0\sigma"'] .
\end{tikzcd}$$

Now, for the nerve of a category this restriction map is in fact an bijection for each n, a property we call the **Segal condition**.

$$\mathcal{N}\mathcal{C}_n \cong \mathcal{N}\mathcal{C}_1 \underset{\mathcal{N}\mathcal{C}_0}{\times} \cdots \underset{\mathcal{N}\mathcal{C}_0}{\times} \mathcal{N}\mathcal{C}_1 = \operatorname{Hom}\mathcal{C} \underset{\operatorname{Ob}\mathcal{C}}{\times} \cdots \underset{\operatorname{Ob}\mathcal{C}}{\times} \operatorname{Hom}\mathcal{C} \qquad \text{(Segal condition for } \mathcal{N}\mathcal{C}\text{)}$$

This is witnessed by the fact that any composite of morphisms is unique for a category. In dimension 2 it is witnessed by the fact that a composite of two morphisms is unique. And in dimension 3 we may build up a horn of the 3-simplex from its spine in two distinct ways by taking composites of consecutive edges. This corresponds to the fact that given three morphisms $x \xrightarrow{f} y \xrightarrow{g} z \xrightarrow{h} w$ we may either take the composite $h \circ (g \circ f)$ or the composite $(h \circ g) \circ f$, hence uniqueness is witnessed by associativity of the composition operation. All higher-order conditions follow from these. In fact, building up horns in this way gives us an alternative equivalent condition:

Proposition 7.14. *The restriction maps* $\operatorname{Hom}_{\mathbf{sSet}}(\Delta^n, X) \to \operatorname{Hom}_{\mathbf{sSet}}(\mathrm{Sp}^n, X)$ *are bijections for all n if and only if the restriction maps* $\operatorname{Hom}_{\mathbf{sSet}}(\Delta^n, X) \to \operatorname{Hom}_{\mathbf{sSet}}(\Lambda^n_k, X)$ *are bijections for the inner horns, i.e. all n and k where $0 < k < n$.*

Proof. The Segal condition yields the bijections for all inner horns immediately since for $0 < k < n$ we have an inclusion $\mathrm{Sp}^n \hookrightarrow \Lambda^n_k$ which we may restrict along. The converse may be shown inductively by building up inner horns from spines. For a full proof see for instance [Lur09, Proposition 1.1.2.2]. □

Remark 7.15. We should not expect the inner horn condition to hold for the outer horns. For instance, in the case of $n = k = 2$, this would imply that every diagram of the following form admit a completion

$$\begin{tikzcd} & y & \\ y \arrow[ur, equal] \arrow[rr, dashed] & & x \arrow[ul, "f"'] \end{tikzcd}$$

i.e. that all morphisms are right invertible.

We note that not all simplicial sets satisfy the Segal condition:

Example 7.16 (Non-example for Segal condition)**.** The simplicial set X generated by the following data

$$X_0 = \{v\} \quad X_1 = \{s_0(v)\} \quad X_2 = \{\sigma, s_0^2(v)\} \quad d_i(\sigma) = s_0(v),$$

which may be envisioned as a sphere with a distinguished point, does not satisfy the Segal condition for $n = 2$. Observe that $X_1 \times_{X_0} X_1 = \{(s_0(v), s_0(v))\}$ has cardinality 1 while X_2 has cardinality 2 (recall from Lemma 7.2 that $s_1 s_0 = s_0 s_0$). Hence we may also conclude that there is no (1-)category which "looks like" a sphere.

However, the Segal condition is enough to ensure that our simplicial set is the nerve of a category. Let $\mathbf{sSet}_{\mathrm{Segal}}$ denote the full subcategory of $\mathbf{sSet}$ whose objects are simplicial sets which satisfy the Segal condition.

Theorem 7.17 ([Ras18, Theorem 1.27 and 1.28])
The nerve functor restricted to $\mathbf{sSet}_{\mathrm{Segal}}$ *is an equivalence.*

Proof. See [Ras18, Theorem 1.27 and 1.28]. □

7.4 Kan complexes

We take a quick detour to discuss groupoids, categories in which all morphisms are invertible, in the context of the nerve construction. This is for the sake of introducing Kan complexes which give a model of ∞-groupoids in simplicial sets.

We're perhaps not doing the groupoids justice by describing them as particular categories. It was suggested by Vladimir Voevodsky that groupoids are more fundamental to mathematics than categories. As he writes in [Voe14], it is not categories, but groupoids which are "sets in the next dimension." While categories are in fact "*partially ordered sets* in the next dimension." This was one of the key insights that led him to consider homotopy-theoretic foundations rather than category-theoretic ones. As he remarked, not all natural mathematical constructions are functorial. However, all meaningful mathematical constructions are preserved under their appropriate notion of equivalence. Ulrik Buchholtz in [Buc18] gives the example of the commutative center of a group. This is arguably a natural construction to perform on a group, but it is not functorial. It is however preserved under equivalences: isomorphic groups have isomorphic centers.

The following table suggests a general scheme.

Dimension	Sets	Partial orders	Spaces
0	sets	partial orders	discrete spaces
1	groupoids	categories	1-truncated spaces
2	2-groupoids	$(2,1)$-categories	2-truncated spaces
n	n-groupoids	$(n,1)$-categories	n-truncated spaces
∞	∞-groupoids	$(\infty,1)$-categories	spaces

For any given category there are two canonical groupoids to consider, respectively constructed as the left and right adjoints to the inclusion functor of groupoids into categories. These are called the groupoidal localization and core of the category respectively.

$$\begin{tikzcd} \mathbf{Gpd} \arrow[r, hook] & \mathbf{Cat} \arrow[l, bend right=50, "\mathrm{Loc}"', "\perp"] \arrow[l, bend left=50, "\mathrm{Core}", "\perp"'] \end{tikzcd}$$

Explicitly, the core of a category $\mathcal{C}$ is characterized as the maximal subgroupoid contained in $\mathcal{C}$, while the groupoidal localization of $\mathcal{C}$ is the minimal groupoid containing $\mathcal{C}$. For instance, the core of the free-standing morphism category $\mathbb{2}$ (earlier referred to as $[1]$) is the discrete two-object category $\mathbf{2}$, while the groupoidal localization is the free-standing isomorphism category $\mathbb{I}$ (the indiscrete two-object category if you will).

Remark 7.18 (Classifying space)**.** Interpreted as spaces, categories are equivalent to their groupoidal localizations. By this, we mean that they are weakly homotopy equivalent under the geometric realization of their nerve

$$\mathcal{B}\mathcal{C} := |\mathcal{N}\mathcal{C}|.$$

This composite construction is called the **classifying space** of the category.

We may characterize nerves of groupoids similarly to how we characterized nerves of categories. Using the horn description as in Proposition 7.14 we note that since all morphisms are invertible in a groupoid we also have bijections of hom-sets at the outer horns Λ_0^2 and Λ_2^2. From this, it follows that we have bijections of hom-sets for *all* horns in dimensions 2 and up:

Proposition 7.19. *A simplicial set X is the nerve of a groupoid precisely if the restriction maps*

$$\mathrm{Hom}_{\mathbf{sSet}}(\Delta^n, X) \to \mathrm{Hom}_{\mathbf{sSet}}(\Lambda_k^n, X)$$

are bijections for all horns Λ_k^n where $n > 1$.

If we relax this condition to only requiring these restriction maps to have left inverses, we get the notion of a *Kan complex*:

Definition 7.20 (Kan Complexes)**.** A simplicial set X is a **Kan complex** if the restriction maps

$$\mathrm{Hom}_{\mathbf{sSet}}(\Delta^n, X) \to \mathrm{Hom}_{\mathbf{sSet}}(\Lambda_k^n, X) \qquad \text{(Kan condition)}$$

are surjections.[13]

This relaxation ensures that every composite is still witnessed by some filler, but now this filler may no longer be unique. It may however relate to other fillers by even higher fillers and so on.

Example 7.21. For any topological space X, the singular simplicial complex $S_\bullet(X)$ is a Kan complex.

Example 7.22. The standard simplexes Δ^n are not Kan complexes except at $n = 0$.

Kan complexes form a model of ∞-groupoids in simplicial sets. Using the adjunction between geometric realization and singular complexes it may be shown that they model the homotopy types of spaces. Hence we are justified in calling Kan complexes simply

[13] In constructive mathematics this condition is strictly weaker than the condition that these restriction maps admit left inverses, i.e. sections. Their equivalence requires the axiom of choice.

spaces. Moreover, Kan complexes model the types of homotopy type theory [KL18] creating a direct link between the type theory and classical homotopy theory. Lastly, Kan complexes are the underlying objects in the theory of simplicial spaces, which gives us a model for the homotopy theory of $(\infty, 1)$-categories called Rezk spaces. All of this will be made more precise in the course of the next sections using the powerful language of model categories.

8 Homotopical categories

In this section, we elaborate on what we may in the most general terms consider a homotopy theory. Unlike theories such as topology or category theory, which can be concisely and completely described as the study of certain kinds of structures and certain kinds of maps, homotopy theory has taken on a multitude of meanings throughout its lifetime and does not in its full generality refer to the study of any one given structure. Classically, it has meant the study of homotopies in topology: paths between points in spaces. But homotopy theory has grown to encompass much more than just this.

One reoccurring problem in homotopy theory is the problem of weak identification. We are given some weak system of identification in a theory, which we call homotopy equivalences. We wish to consider the structures in this theory only *up to* the given system of identification, although the theory itself may express properties richer than what is recognized by them. This naturally leads us to what we call the theory's **homotopy theory**.

In this setting what we mean by a theory is just any category, and the weak system of identification will mean a specified subclass of morphisms. We call a category equipped with such a subclass of morphisms a **homotopical category**. It is a remarkable fact that in such a general setting as this we recover many notions recognizable from homotopy theory. In particular, we will define localizations, homotopical functors, derived functors, and homotopy limits.

We restrict ourselves to the special case where the distinguished subclass of morphisms satisfy a closedness property, although everything mentioned in this section is more general. This is because the literature is most developed with regards to this closedness property, and generalizing poses no benefit to us in this thesis.

Definition 8.1 (Category with weak equivalences)**.** A category with weak equivalences is a category $\mathcal{C}$ with a subclass of distinguished morphisms $\mathtt{W}$ containing the isomorphisms and satisfying the **2-out-of-3** property: given a commutative triangle in $\mathcal{C}$ (using bullets as anonymous labels for the vertices)

$$\begin{tikzcd} & \bullet \arrow[dr, "g"] & \\ \bullet \arrow[ur, "f"] \arrow[rr, "g\circ f"] & & \bullet \end{tikzcd},$$

if two of its edges are in $\mathtt{W}$ then so is the third.

The 2-out-of-3 property is one out of many closedness properties we could ask of our weak equivalences. What all of the closedness properties have in common is that they generalize some aspect of being an isomorphism.

Lemma 8.2. *A class of weak equivalences is a wide subcategory.*

Proof. It is closed under composition by the 2-out-of-3 property, and since it contains the isomorphisms it, in particular, contains the identities on all objects. □

Lemma 8.3. *Given a category with weak equivalences* $(\mathcal{C}, \mathtt{W})$ *and a small category* $\mathcal{J}$*, then we may canonically endow the functor category* $\mathcal{C}^{\mathcal{J}}$ *with the weak equivalences* $\mathtt{W}^{\mathcal{J}}$*.*

Proof. Natural transformations are determined degreewise. □

Sometimes weak equivalences are characterized by being "inverted by a functor" $F : \mathcal{C} \to \mathcal{A}$ in the sense that they are exactly the morphisms of $\mathcal{C}$ mapped to isomorphisms in $\mathcal{A}$. In other words, that the following is a pullback square

$$\begin{tikzcd} \mathtt{W} \arrow[r, "F|_{\mathtt{W}}"] \arrow[d, hook] \arrow[dr, phantom, "\lrcorner", very near start] & \operatorname{Core} \mathcal{A} \arrow[d, hook] \\ \mathcal{C} \arrow[r, "F"] & \mathcal{A}\,. \end{tikzcd}$$

Example 8.4 (Examples of weak equivalences)**.** Some common examples of weak equivalences and their inverting functors:

(i) Isomorphisms in any category are inverted by the identity functor. We may call these the **discrete equivalences**.

(ii) All morphisms in any category are inverted by mapping to the terminal category $\mathcal{C} \to \mathbf{1}$. We call these the **indiscrete equivalences**.

(iii) Bijections in **Top** are inverted by the underlying set functor.

(iv) Homotopy equivalences in **Top** are inverted by transporting to the homotopy category $\mathbf{h}(-) : \mathbf{Top} \to \mathbf{hTop}$.

(v) Weak homotopy equivalences in **Top** are inverted by the fundamental ∞-groupoid functor $\Pi_\infty(-)$, which can for instance be taken to be the singlular complex functor $\mathrm{S}_\bullet(-)$ using the model of ∞-groupoids in Kan complexes.

(vi) Chain homotopy equivalences in $\mathrm{Ch}_\bullet \mathcal{A}$ for $\mathcal{A}$ an additive category are inverted by transporting to the homotopy category $\mathbf{h}(-) : \mathrm{Ch}_\bullet \mathcal{A} \to \mathbf{h}\mathrm{Ch}_\bullet \mathcal{A}$.

(vii) Quasi-isomorphisms in $\mathrm{Ch}_\bullet \mathcal{A}$ for $\mathcal{A}$ an abelian category are inverted by taking homology $H_\bullet(-)$.

(viii) Weak homotopy equivalences in **sSet**, meaning maps whose geometric realization is a weak homotopy equivalence in **Top**.

(ix) Homotopy equivalences in **sSet**, meaning maps of simplicial sets $f : A \to B$ for which there exists a map $g : B \to A$ and *homotopies* $p : \Delta^1 \times A \to A$ and $q : \Delta^1 \times B \to B$ such that p restricts to id_A and $g \circ f$ on $\{0\} \times A \cong A$ and $\{1\} \times A \cong A$ respectively, and q restricts to $f \circ g$ and id_B on the respective endpoints.

Given Example 8.4 (ii), we cannot hope to prove too many interesting properties of weak equivalences by themselves. It is however a basic idea on which we build more advanced constructions, which we turn our attention to next.

Definition 8.5 (Homotopical functor)**.** A functor $F : \mathcal{C} \to \mathcal{D}$ between categories with weak equivalences is said to be **homotopical** if it maps weak equivalences to weak equivalences

$$\begin{tikzcd} \mathtt{W} \arrow[r, "F|_{\mathtt{W}}"] \arrow[d, hook] & \mathtt{V} \arrow[d, hook] \\ \mathcal{C} \arrow[r, "F"] & \mathcal{D}. \end{tikzcd}$$

Example 8.6 (Examples of homotopical functors)**.**

(i) Any functor whose domain is discrete or codomain is indiscrete.

(ii) The geometric realization functor from simplicial sets to topological spaces sends weak homotopy equivalences to weak homotopy equivalences by definition.

(iii) The functor sending topological spaces to their singular sets in **sSet** sends weak homotopy equivalences to weak homotopy equivalences.

For a category with weak equivalences $\mathtt{W} \subseteq \mathcal{C}$, we may construct a category $\mathcal{C}[\mathtt{W}^{-1}]$ where the morphisms in $\mathtt{W}$ have been formally inverted. This can for instance be done using a *calculus of fractions* [GZ67]. Such a category satisfies a certain universal property, known as being the **localization of $\mathcal{C}$ at $\mathtt{W}$**:

Definition 8.7 (Localization)**.** The **localization of $\mathcal{C}$ at $\mathtt{W}$** is a category denoted $\mathcal{C}[\mathtt{W}^{-1}]$, or $\mathbf{h}\mathcal{C}$ when the specified class of weak equivalences is implicit, enjoying the universal property that for any category $\mathcal{A}$, the category of homotopical functors $(\mathcal{C}, \mathtt{W}) \to (\mathcal{A}, \operatorname{Core} \mathcal{A})$ is isomorphic to the category of functors $\mathcal{C}[\mathtt{W}^{-1}] \to \mathcal{A}$ as fibers over $\mathcal{C} \to \mathcal{A}$ in $\mathcal{C}/\mathbf{Cat}$:

$$\begin{tikzcd} \mathtt{W} \arrow[r, dashed] \arrow[d, hook] & \operatorname{Core} \mathcal{A} \arrow[d, hook] \\ \mathcal{C} \arrow[r, dashed] & \mathcal{A} \end{tikzcd} \quad \cong \quad \mathcal{C}[\mathtt{W}^{-1}] \dashrightarrow \mathcal{A}.$$

Applying this correspondence to the identity functor at $\mathcal{C}[\mathtt{W}^{-1}]$, we get a localization functor $\iota : \mathcal{C} \to \mathcal{C}[\mathtt{W}^{-1}]$ such that the following triangles commute

$$\begin{tikzcd} (\mathcal{C}[\mathtt{W}^{-1}] \to \mathcal{A}) \arrow[rr, "\cong"] \arrow[dr, hook, "- \circ \iota"] & & ((\mathcal{C}, \mathtt{W}) \to (\mathcal{A}, \operatorname{Core} \mathcal{A})). \arrow[dl, hook] \\ & (\mathcal{C} \to \mathcal{A}) & \end{tikzcd}$$

When the class of weak equivalences is taken implicitly, we call this category the **homotopy category** of $\mathcal{C}$.

For a homotopical functor $F : \mathcal{C} \to \mathcal{D}$ there is an induced functor between homotopy categories since postcomposing with the universal functor $\mathcal{D} \to \mathbf{h}\mathcal{D}$ sends weak equivalences in $\mathcal{C}$ to isomorphisms in $\mathbf{h}\mathcal{D}$. When the functor F is not homotopical, however, the best we can do is take a universal approximation.

Definition 8.8 (Absolute Kan extension)**.** A left Kan extension (L, λ) of F along G,

$$\begin{tikzcd} \mathcal{A}' \arrow[dr, dashed, "L"] & \\ \mathcal{A} \arrow[u, "G"] \arrow[r, "F"'] \arrow[ur, phantom, "\lambda\Uparrow"] & \mathcal{B}, \end{tikzcd}$$

is said to be **absolute** if the whiskered composite along any functor $H : \mathcal{B} \to \mathcal{C}$, $(HL, H\lambda)$, is a left Kan extension of HF along G.

Definition 8.9 (Derived functors)**.** A **left derived functor** of $F : \mathcal{C} \to \mathcal{D}$ is a homotopical functor $\mathbb{L}F : \mathcal{C} \to D$ equipped with a natural transformation $\lambda : \mathbb{L}F \Rightarrow F$ such that $(d\mathbb{L}F, d\lambda)$ defines an absolute right Kan extension of dF along c.

$$\begin{tikzcd} \mathcal{C} \arrow[r, bend left, "F"] \arrow[r, bend right, "\mathbb{L}F"'] \arrow[r, phantom, "\Uparrow"] & \mathcal{D} \end{tikzcd} \qquad \begin{tikzcd} \mathcal{C} \arrow[r, "F"] \arrow[d, "c"'] & \mathcal{D} \arrow[d, "d"] \\ \mathbf{h}\mathcal{C} \arrow[r, "d\mathbb{L}F"'] \arrow[ur, phantom, "d\lambda \nearrow"] & \mathbf{h}\mathcal{D} \end{tikzcd}$$

If the left derived functor $\mathbb{L}F$ exists, then the **total left derived functor**

$$\mathbf{L}F := d\mathbb{L}F : \mathbf{h}\mathcal{C} \to \mathbf{h}\mathcal{D}$$

exists by the universal property of localization. The coduals[14] of these constructions are the functor's **right derived functor** and **total right derived functor**.

Particularly central examples of derived functors, when they exist, are the homotopy limit and homotopy colimit functors.

Definition 8.10 (Homotopy (co)limit, global)**.** Given that all ordinary limits of shape $\mathcal{J}$ in $\mathcal{C}$ exist, we have a limit functor lim which is right adjoint to the (already homotopical) diagonal functor Δ. This functor takes diagrams $\mathcal{C}^{\mathcal{J}}$ to their limits in $\mathcal{C}$. The right derived functor of lim, if it exists, defines the **homotopy limit** functor

$$\text{holim} := \mathbb{R}\text{lim} : \mathcal{C}^{\mathcal{J}} \to \mathcal{C},$$

which picks out the **homotopy limit** of a diagram. Dually, if all colimits of shape $\mathcal{J}$ exist, we define the **homotopy colimit functor** as

$$\text{hocolim} := \mathbb{L}\text{colim} : \mathcal{C}^{\mathcal{J}} \to \mathcal{C}.$$

[14]By **co**dual, we mean the dual construction where the 1-arrows' orientations remain unchanged, but the 2-arrows are reversed.

Homotopy limits are generally not limits in the homotopy category, but rather limits universal up to homotopically coherent cones. In particular, their total derived counterparts are functors $\mathbf{h}(\mathcal{C}^{\mathcal{J}}) \to \mathbf{h}\mathcal{C}$, not functors $(\mathbf{h}\mathcal{C})^{\mathcal{J}} \to \mathbf{h}\mathcal{C}$.

Remark 8.11 (Simplicial localization)**.** Localizations as presented here suffer from a certain deficiency. Formally *inverting* all weak equivalences in a sense destroys too much information to preserve the full homotopy theory. In particular, one may obtain equivalent homotopy categories from non-equivalent (in the appropriate sense) homotopical categories. To remedy this, we may instead build a homotopy category enriched in simplicial sets, an $(\infty, 1)$-category, called the **simplicial localization** of $\mathtt{W} \subseteq \mathcal{C}$ in which the weak equivalences are only weakly inverted. Instead of introducing formal inverses in the 1-categorical sense, we introduce formal homotopy inverses. Inverses that do not compose to the identity on the nose, but are only witnessed as inverses by higher homotopies. Higher homotopies which again are subject to laws up to even higher homotopies. The simplicial localization may for instance be constructed using the *hammock localization* [DK80]. A remarkable fact is that "every" $(\infty, 1)$-category may be constructed in this way. Hence sometimes $(\infty, 1)$-categories are referred to as homotopy theories (e.g. [Rez01]). We may note that this terminology is consistent with the definition we outlined at the beginning of this section.

9 The lifting problem

A lifting problem may in general refer to a multitude of things. Indeed, we will see many kinds of lifting problems in the course of this thesis alone. Examples are Kan lifts, absolute Kan lifts, absolute lifting diagrams, and perhaps most fundamentally lifts of cospans. What they all have in common is that they pose a question as to the existence of a certain kind of factorization of morphisms.

In this section, however, we inspect lifting problems of one specific shape, referred to as *the lifting problem.*

Definition 9.1 (Lifting problem)**. The lifting problem** between two morphisms l and r is the question of whether a commutative square of the following form admits a diagonal map h maintaining commutativity

$$\begin{tikzcd} \cdot \arrow[r] \arrow[d, "l"'] & \cdot \arrow[d, "r"] \\ \cdot \arrow[r] \arrow[ur, dashed, "h"] & \cdot \end{tikzcd}.$$

If h exists it is called a **lift** or **lifting of** l **against** r. If every lifting problem between l and r admits a lift, then l is said to have the **left lifting property** with respect to r and r is said to have the **right lifting property** with respect to l.

If such liftings are unique, we say that l is **left orthogonal** to r, and that r is **right orthogonal** to l. As an abbreviation, we say that r is l**-orthogonal**.

We say an *object* X has the left lifting/orthogonality property with respect to a morphism r if the initial inclusion $\mathbf{0} \to X$ has this property, and X has the right lifting/orthogonality property with respect to a morphism l if the terminal projection $X \to \mathbf{1}$ has this property. As with morphisms, we say X is l-orthogonal if it is right orthogonal to l.

Example 9.2 (Motivating example for lifting properties)**.** Lifting properties vastly generalize the motivating example of the classical homotopy lifting and homotopy extension properties characterizing respectively fibrations and cofibrations in point-set topology:

$$\begin{tikzcd} A \arrow[r] \arrow[d, tail, "i_0"', "\wr"] & X \arrow[d, "f"] \\ A \times I \arrow[r] \arrow[ur, dashed] & Y \end{tikzcd} \quad \text{and} \quad \begin{tikzcd} X \arrow[r] \arrow[d, tail, "f"'] & A^I \arrow[d, "p_0", "\wr"'] \\ Y \arrow[r] \arrow[ur, dashed] & A. \end{tikzcd}$$

A map of topological spaces $f : X \to Y$ is a fibration precisely if it has the right lifting property with respect to the left end-point inclusion $i_0 : A \cong A \times \{0\} \to A \times I$ for all topological spaces A. And f is a cofibration precisely if it has the left lifting property with respect to evaluation at the left end-point $p_0 : A^I \to A^{\{0\}} \cong A$ for all topological spaces A. For specificity, we refer to these maps as Hurewicz (co)fibrations.

Remark 9.3 (Alternative formulation of lifting problem)**.** The morphism $l \in \mathrm{Hom}(a, b)$ has the left lifting property with respect to $r \in \mathrm{Hom}(c, d)$ if and only if the map that sends lifts to the lifting problem they solve

$$\mathrm{Hom}(b, c) \to \mathrm{Hom}(a, c) \times_{\mathrm{Hom}(a,d)} \mathrm{Hom}(b, d)$$

is a surjection. It has the left orthogonality property if this map is a bijection.

In the case that r is a terminal projection, this alternative formulation has an even simpler form, since the set of lifting problems is just the set of morphisms $\mathrm{Hom}(a, c)$. In this setting, we remark that we may reformulate the Segal condition, Kan condition, and many of their related properties in terms of lifting and orthogonality properties. For instance, the Segal condition is precisely the condition that a simplicial set is orthogonal to the spine inclusions $\mathrm{Sp}^n \hookrightarrow \Delta^n$ for all n.

For the remainder of this section, we fix a category $\mathcal{C}$ with a distinguished class of morphisms M having a hypothesized element m. Furthermore, we define $\mathtt{L}$ and $\mathtt{R}$ to be the classes of morphisms in $\mathcal{C}$ with respectively the left and right lifting property with regard to all morphisms in M, denoted $\mathtt{L} := {}^{\boxslash}M$ and $\mathtt{R} := M^{\boxslash}$.

Our current goal is to establish as many properties as we can about $\mathtt{L}$ and $\mathtt{R}$. Fortunately for us, these classes are formal duals to one another and so a theorem for one immediately yields a dual theorem for the other.

Lemma 9.4. *The morphism classes* $\mathtt{L}$ *and* $\mathtt{R}$ *are closed under composition.*

Proof. By duality we need only consider the case for the morphism class $\mathtt{L}$, so assume given a lifting problem of gf against m for $f, g \in \mathtt{L}$ as the solid part of the following diagram

$$
\begin{tikzcd}
\cdot \arrow[r, "a"] \arrow[d, "f"'] & \cdot \arrow[dd, "m"] \\
\cdot \arrow[ur, dashed] \arrow[d, "g"'] & \\
\cdot \arrow[r, "b"'] \arrow[uur, dotted] & \cdot
\end{tikzcd}.
$$

We attain the diagonal maps by taking the liftings

$$
\begin{tikzcd}
\cdot \arrow[r, "a"] \arrow[d, "f"'] & \cdot \arrow[d, "m"] \\
\cdot \arrow[r, "bg"'] \arrow[ur, dashed, "h"] & \cdot
\end{tikzcd}
\quad \text{and then} \quad
\begin{tikzcd}
\cdot \arrow[r, dashed, "h"] \arrow[d, "g"'] & \cdot \arrow[d, "m"] \\
\cdot \arrow[r, "b"'] \arrow[ur, dotted] & \cdot
\end{tikzcd}
$$

The dotted arrow now solves our original lifting problem. $\square$

Lemma 9.5. *The classes* `L` *and* `R` *contain the isomorphisms.*

Proof. By duality, we only consider the first case. Let f be an isomorphism with inverse f^{-1} and assume we have a lifting problem f against m. Then we have the following diagonal

$$
\begin{tikzcd}
\cdot \arrow[r, "a"] \arrow[d, "f"'] & \cdot \arrow[d, "m"] \\
\cdot \arrow[r, "b"'] \arrow[ur, dashed, "af^{-1}"] & \cdot
\end{tikzcd}.
$$

We may immediately verify that $af^{-1}f = a\,\mathrm{id} = a$. Moreover, commutativity of the outer square states that $bf = ma$, and by precomposing with f^{-1} we get $bff^{-1} = b = maf^{-1}$ which shows that the lower right triangle commutes. Hence we have a lifting and $f \in \mathtt{L}$. $\square$

Corollary 9.6. *The classes* `L` *and* `R` *are wide subcategories of* $\mathcal{C}$*.* $\square$

Having established this basic fact, we move on to show that our lifting classes already enjoy many stability properties out of the box. As we will shortly see, it is true that `L` is stable[15] under finite coproducts, pushouts and arbitrary colimits of cotowers. Dually, `R` is stable under taking finite products, pullbacks, and arbitrary limits of towers.

Lemma 9.7. *The class* `L` *is stable under coproducts and* `R` *is stable under products.*

Proof. By duality it suffices to prove the first claim, so let f and g be members of `L` and suppose we are given a lifting problem of $f \sqcup g$ against m. Then we may adjoin either f or g to the lifting square in the following manner

$$
\begin{tikzcd}
\cdot \arrow[r] \arrow[d, "f"'] & \cdot \arrow[r] \arrow[d, "f\sqcup g"'] & \cdot \arrow[d, "m"] \\
\cdot \arrow[r] \arrow[urr, dashed, "f'"] & \cdot \arrow[r] & \cdot
\end{tikzcd}
\quad \text{and} \quad
\begin{tikzcd}
\cdot \arrow[r] \arrow[d, "g"'] & \cdot \arrow[r] \arrow[d, "f\sqcup g"'] & \cdot \arrow[d, "m"] \\
\cdot \arrow[r] \arrow[urr, dashed, "g'"] & \cdot \arrow[r] & \cdot
\end{tikzcd}
$$

yielding the dashed lifts. Hence by the coproduct property we have a unique solution to the original lifting problem. $\square$

[15] By "*being stable under*" we mean that if the diagram admits a (co)limit in the ambient category, then this is an inhabitant of our distinguished subclass.

Lemma 9.8. *The class* $\mathtt{L}^2$ *is stable under pushout and* $\mathtt{R}^2$ *is stable under pullback considered as objects of* $\mathcal{C}^2$*.*

Proof. By duality we need only consider the first case. So assume given an $l \in \mathtt{L}$ and f its pushout. Moreover assume we have a lifting problem of f against m.

$$\begin{tikzcd}
\cdot \arrow[r] \arrow[d, "l"'] & \cdot \arrow[r] \arrow[d, "f"] & \cdot \arrow[d, "m"] \\
\cdot \arrow[r] \arrow[urr, dashed] & \cdot \arrow[r] \arrow[ur, dotted] \arrow[ul, phantom, "\ulcorner", very near start] & \cdot
\end{tikzcd}$$

The dashed arrow exists by the lifting property between l and m, hence the dotted arrow exists by the pushout property and solves the lifting problem. $\square$

Lemma 9.9. *The class* $\mathtt{L}$ *is stable under taking colimits of arbitrary cotowers and* $\mathtt{R}$ *is stable under taking limits of arbitrary towers.*

Proof. Again by duality we need only consider the case for $\mathtt{L}$. So assume given a cotower diagram $X : \mathtt{L}^\omega$ for some limit ordinal ω with a colimit X_ω and limiting composite f. Moreover assume given a lifting problem of f against m

$$\begin{tikzcd}
X_0 \arrow[r, "a"] \arrow[d, "f"'] & A \arrow[d, "m"] \\
X_\omega \arrow[r, "b"'] & B\,.
\end{tikzcd}$$

By well-foundedness of the ordinal we may for any $i \in \omega$ factor f as the composite $X_0 \xrightarrow{X(0 \le i)} X_i \xrightarrow{f_i} X_\omega$. Hence in particular since $X(0 \le i) \in \mathtt{L}$ we have lifting squares

$$\begin{tikzcd}
X_0 \arrow[r, "a"] \arrow[d, "X(0\le i)"'] & A \arrow[d, "m"] \\
X_i \arrow[r, "bf_i"'] \arrow[ur, dashed] & B\,,
\end{tikzcd}$$

But by commutativity these diagonal maps constitute a cocone under X and so by the universal property of the colimit we get a solution to our original lifting problem. $\square$

9.1 Weak factorization systems

In this section, we introduce the notion of weak factorization systems building on the notion of lifting properties introduced in the previous section. These were not in Quillen's original vocabulary, but have proven to be a useful abstraction in defining model structures. Following [Rie09] we will also prove retract stability, a property some authors take as an axiom (e.g. [DS95]).

Definition 9.10 (Weak factorization system)**.** A **weak factorization system**[16] on a category $\mathcal{C}$ is a pair of distinguished classes of morphisms $\mathtt{L}$ and $\mathtt{R}$ such that

[16] We say the factorization system is weak because the lifting morphisms and factorizations are not generally unique nor do they have any functorial properties. These properties would respectively yield *orthogonal* and *functorial* factorization systems.

[Fac] Any morphism f may be factored as a morphism in $\mathtt{L}$ followed by a morphism in $\mathtt{R}$

$$\begin{tikzcd} & \cdot \arrow[dr, "r"] & \\ \cdot \arrow[ur, "l"] \arrow[rr, "f"] & & \cdot \end{tikzcd}.$$

[LLC] $\mathtt{L}$ is the class of morphisms with the left lifting property with respect to $\mathtt{R}$.

[RLC] $\mathtt{R}$ is the class of morphisms with the right lifting property with respect to $\mathtt{L}$.

Remark 9.11 (Self-duality)**.** The notion of a weak factorization system is self-dual.

Example 9.12. Some natural examples of factorization systems.

(i) The monomorphisms and epimorphisms in **Set** or vice versa.

(ii) n-connected and n-truncated maps in **Top**.

(iii) Functors bijective on objects and functors bijective on hom-sets (i.e. fully faithful) in **Cat**.

All of these examples satisfy the additional property of being *orthogonal* factorization systems, meaning that the factorizations and liftings are essentially unique. This condition is too strict for the kind the factorization systems we are interested in with relation to model categories, but to motivate them takes a little more effort.

Lemma 9.13. *Given a weak factorization system* $(\mathtt{L}, \mathtt{R})$ *on* $\mathcal{C}$*, the intersection* $\mathtt{L} \cap \mathtt{R}$ *is exactly the isomorphisms in* $\mathcal{C}$*.*

Proof. By Lemma 9.5 we already know the isomorphisms are contained in $\mathtt{L} \cap \mathtt{R}$. So only the converse remains.

Let f be in $\mathtt{L} \cap \mathtt{R}$. Then it admits a lifting against itself

$$\begin{tikzcd} \cdot \arrow[r, equal] \arrow[d, "f"'] & \cdot \arrow[d, "f"] \\ \cdot \arrow[r, equal] \arrow[ur, dashed, "g"] & \cdot \end{tikzcd}$$

satisfying $gf = \mathrm{id}$ and $fg = \mathrm{id}$. □

Definition 9.14 (Retract)**.** Let the following be a commutative diagram in a category

$$\begin{tikzcd} A \arrow[r, "s"] \arrow[rr, equal, bend right] & B \arrow[r, "r"] & A \end{tikzcd},$$

then we say A is a **retract** of B, s is a **section** of r and a **split monomorphism**, and r is a **retraction** of s and a **split epimorphism**.

Proposition 9.15 (Retract stability)**.** *The arrow subclasses* $\mathtt{L}^2$ *and* $\mathtt{R}^2$ *are stable under retracts.*

Proof. By duality, we need only show that $\mathtt{L}^2$ is stable under retracts. So assume given a retract f of l and that we have a lifting problem between f and r. By the factorization property of $(\mathtt{L},\mathtt{R})$ we may factor f as $f = f_{\mathtt{R}} f_{\mathtt{L}}$ where $f_{\mathtt{L}} \in \mathtt{L}$ and $f_{\mathtt{R}} \in \mathtt{R}$. Now we may form the following diagram by slight rearrangements: the rightmost column is the lifting problem and the left two columns is the retract of f.

$$\begin{tikzcd}
\cdot \arrow[r, "j"] \arrow[rr, bend left, equal] \arrow[dd, "f"'] & \cdot \arrow[r, "q"] \arrow[d, equal] & \cdot \arrow[r, "a"] \arrow[d, "f_{\mathtt{L}}"] & \cdot \arrow[d, "r"] \\
 & \cdot \arrow[r, "f_{\mathtt{L}} q"] \arrow[d, "l"] & \cdot \arrow[r, "b f_{\mathtt{R}}"] \arrow[d, "f_{\mathtt{R}}"] \arrow[ur, dashed, "y"] & \cdot \arrow[d, equal] \\
\cdot \arrow[r, "i"] \arrow[rr, bend right, equal] & \cdot \arrow[r, "p"] \arrow[ur, dashed, "x"] & \cdot \arrow[r, "b"] & \cdot
\end{tikzcd}$$

This yields a lift x of l against $f_{\mathtt{R}}$ and a lift y of $f_{\mathtt{L}}$ against r. Let ϕ be the composition $\phi = yxi$. Then by commutativity we have $\phi f = yxig = yxlj = y f_{\mathtt{L}} qj = aqj = a$ and $r\phi = ryxi = bf_{\mathtt{R}}xi = bpi = b$. Hence ϕ is a lift of f against r. □

Another connection between retracts and lifting properties is the following.

Lemma 9.16 (Retract argument)**.** *Assume $l = r \circ f$ has the left lifting property with respect to r. Then l is a retract of f.*

Proof. We use the lifting property of the below left-hand square to obtain a retraction as displayed below right

$$\begin{tikzcd}
\cdot \arrow[r, "f"] \arrow[d, "l"'] & \cdot \arrow[d, "r"] \\
\cdot \arrow[r, equal] \arrow[ur, dashed, "h"] & \cdot
\end{tikzcd}
\quad \rightsquigarrow \quad
\begin{tikzcd}
\cdot \arrow[r, equal] \arrow[d, "l"'] & \cdot \arrow[r, equal] \arrow[d, "f"] & \cdot \arrow[d, "l"] \\
\cdot \arrow[r, "h"] \arrow[rr, bend right, equal] & \cdot \arrow[r, "r"] & \cdot
\end{tikzcd} .$$

□

Corollary 9.17. *Either class of a weak factorization system determines the other.*

Proof. A map f with the left lifting property with respect to $\mathtt{R}$, may be factorized using the factorization property as $l \circ r$, where $l \in \mathtt{L}$ and $r \in \mathtt{R}$. Hence by the retract argument f is a retract of l and so lives in $\mathtt{L}$ by retract stability. □

To end this section we quote the small object argument, originally due to Quillen. Most notably for us, it allows us to define weak factorization systems on cocomplete categories in terms of a class of morphisms satisfying a small object condition, although it has a multitude of formulations and consequences. To state it we first need to define the notion of cell complexes.

Definition 9.18 (Cell complex)**.** Let $\mathcal{C}$ be a cocomplete category. Then we define the **M-cell complexes** $\operatorname{cell} M$ to be the class of morphisms in $\mathcal{C}$ obtained as transfinite compositions of pushouts and coproducts of morphisms in M.

Example 9.19. Every simplicial set is a cell complex relative to the monomorphisms.

Theorem 9.20 (Small object argument)
Let $\mathcal{C}$ be a cocomplete category and assume that the domains of the morphisms in M are small in an appropriate sense.[17] *Then every morphism in $\mathcal{C}$ factorizes as a morphism in $\operatorname{cell} M$ followed by a morphism in $M^{\boxslash}$ and so in particular $({}^{\boxslash}(M^{\boxslash}), M^{\boxslash})$ defines a weak factorization system in $\mathcal{C}$ which we say is **cofibrantly generated** by M.*

Proof. See for instance [Joy08, D.2]. □

10 Model categories

We are now ready to define model categories. These present an efficient homotopy theory of the underlying homotopical category via the usage of classes of *cofibrations* and *fibrations* defined using weak factorization systems. Model categories require the underlying category to be bicomplete (complete and cocomplete) and as a consequence only present bicomplete $(\infty, 1)$-categories. We do not use them to study ∞-groupoids or $(\infty, 1)$-categories directly however, but rather to study the $(\infty, 1)$-category of ∞-groupoids and the $(\infty, 1)$-category of $(\infty, 1)$-categories.

Instances of categories that admit the structure of a model category include the category of simplicial sets, the category of topological spaces, of groupoids, of categories themselves, and the category of sets. In fact, all of these admit multiple non-equivalent model structures, presenting multiple homotopy theories of the same objects. For instance, topological spaces admit models of strong and of weak homotopy types. Of prime interest to us is the Kan–Quillen model structure on simplicial sets, which models the homotopy theory of ∞-groupoids in which Kan complexes are the fibrant objects.

In the first subsection, we define model structures and model categories, introduce the basic terminology and give a series of examples of interesting model categories.

We then move on to defining (co)fibrant replacement constructions and homotopies of morphisms in a model structure using path and cylinder objects. This leads us to the definition of the homotopy category of a model category giving us an alternative, more tractable construction of the localization of the underlying homotopical category.

In the final subsection, we define the Reedy structure on indexing-categories and demonstrate how one may inductively define model structures on presheaf categories on these taking values in model categories.

Our main sources are [Rie20] and [DS95], although we occasionally use material from [Joy08, App. D and E], [Lur09, App. A] [RV21, App. C], and [Rie09].

10.1 The model structure

There are many interesting hypotheses we can place on a category and its model structure.[18] We use a slightly non-standard definition for model structures, separating them

[17] For instance κ-compact for some regular ordinal κ.

[18] In fact, there is a whole hierarchy of structures one can ask for on the weak factorization systems in of themselves. See for instance [nLabb].

from their underlying category. We place no criteria on the category itself when defining a model structure, but reserve the title of a “model category” for categories that are both endowed with a model structure and satisfy the appropriate completeness criteria.

Definition 10.1 (Model structure)**.** A **model structure** on a category $\mathcal{C}$ is a class of weak equivalences $\mathtt{W}$ as in Definition 8.1 together with two additional classes of morphisms called the **fibrations** $\mathtt{F}$ and **cofibrations** $\mathtt{C}$ such that the pairs $(\mathtt{C}, \mathtt{W} \cap \mathtt{F})$ and $(\mathtt{C} \cap \mathtt{W}, \mathtt{F})$ form weak factorization systems.

A little bit of terminology and notation: we call members of $\mathtt{F} \cap \mathtt{W}$ **acyclic fibrations** and members of $\mathtt{C} \cap \mathtt{W}$ **acyclic cofibrations**. Given a model structure, we will use $\xrightarrow{\sim}$ to denote weak equivalences, $\twoheadrightarrow$ to denote fibrations, and $\hookrightarrow$ to denote cofibrations. For morphisms belonging to multiple classes, we will use superpositions of these. Moreover, if the unique map from an object to the terminal object is a fibration, we say that object is **fibrant**. Dually, we say the object is **cofibrant** if the unique map from the initial object to it is a cofibration. An object which is both fibrant and cofibrant is called **bifibrant**.

Remark 10.2 (Self-duality)**.** The notion of a model structure is self-dual. If $(\mathtt{W}, \mathtt{F}, \mathtt{C})$ is a model structure on $\mathcal{M}$, then $(\mathtt{W}^{\mathrm{op}}, \mathtt{C}^{\mathrm{op}}, \mathtt{F}^{\mathrm{op}})$ is a model structure on $\mathcal{M}^{\mathrm{op}}$.

Theorem 10.3
If $\mathcal{M}$ either admits all pushouts of morphisms in $\mathtt{C} \cap \mathtt{W}$ along sections or admits all pullbacks of morphisms in $\mathtt{F} \cap \mathtt{W}$ along retractions, then the class of weak equivalences is closed under retracts.

Proof adapted from [Joy08, E.1.3]. We write out the proof only under the first completeness hypothesis.

Assume given a weak equivalence w with a retract f. Suppose first that f is a fibration. Now factor w as $w = vu$ using either weak factorization system, then by the 2-out-of-3 property we have $u \in \mathtt{C} \cap \mathtt{W}$ and $v \in \mathtt{F} \cap \mathtt{W}$. We obtain the dashed arrows as shown

$$
\begin{tikzcd}
\bullet \arrow[r, "a"] \arrow[rr, bend left, equal] \arrow[dd, two heads, "f"'] \arrow[dr, dashed, "ua"'] & \bullet \arrow[r] \arrow[d, tail, "u"', "\wr"] & \bullet \arrow[dd, two heads, "f"] \\
& \bullet \arrow[d, two heads, "v"', "\wr"] \arrow[ur, dashed, "t"'] & \\
\bullet \arrow[r] \arrow[rr, bend right, equal] & \bullet \arrow[r] & \bullet
\end{tikzcd}
$$

where t is a lifting of u against f. The top triangles commute, so $tua = \mathrm{id}$, which means that f is a retract of v. As $v \in \mathtt{F} \cap \mathtt{W}$ by Proposition 9.15, f is as well and we are done.

Now let us start again with no hypotheses on f. Factor f as $f = hg$ with $g \in \mathtt{C} \cap \mathtt{W}$

and $h \in \mathtt{F}$ and construct the indicated pushout which is admitted by hypothesis

$$\begin{tikzcd}
\bullet \arrow[r] \arrow[d, tail, "g"', "\wr"] \arrow[rr, bend left, equal] & \bullet \arrow[r, "e"] \arrow[d, tail, "c"', "\wr"] \arrow[dd, bend left, "w"] & \bullet \arrow[d, tail, "g"', "\wr"] \\
\bullet \arrow[r] \arrow[d, "h"'] & \bullet \arrow[ul, phantom, "\ulcorner", very near start] \arrow[r, dashed, "y"] \arrow[d, dashed, "d"', "\wr"] & \bullet \arrow[d, "h"'] \\
\bullet \arrow[r, "b"] \arrow[rr, bend right, equal] & \bullet \arrow[r] & \bullet
\end{tikzcd}.$$

By Lemma 9.8 the left class of a weak factorization system is stable under pushouts, so $c \in \mathtt{C} \cap \mathtt{W}$. The arrows w and bh form a cone over the pushout square, so there is a unique morphism d as shown such that $dc = w$. By the 2-out-of-3 property, $d \in \mathtt{W}$. Similarly, ge and id form a cone over the pushout square, so there is a unique morphism y as shown above such that $yx = \mathrm{id}$. The lower two squares now display h as a retract of d. As $d \in \mathtt{W}$ and $h \in \mathtt{F}$, the previous argument shows that $h \in \mathtt{W}$. But since g is in $\mathtt{W}$, by the 2-out-of-3 property we have $f = hg \in \mathtt{W}$. Thus $\mathtt{W}$ is stable under retracts. □

A model structure is overdetermined by its classes of cofibrations, fibrations and weak equivalences.

Theorem 10.4
A model structure on a bicomplete category is uniquely determined by any of the following data:

(i) Two out of the three classes $\mathtt{C}$, $\mathtt{F}$, $\mathtt{W}$.

(ii) The fibrant objects and either $\mathtt{C}$ *or* $\mathtt{F} \cap \mathtt{W}$.

(iii) The cofibrant objects and either $\mathtt{F}$ *or* $\mathtt{C} \cap \mathtt{W}$.

(iv) The bifibrant objects and any of the classes $\mathtt{C}$, $\mathtt{C} \cap \mathtt{W}$, $\mathtt{F}$, *or* $\mathtt{F} \cap \mathtt{W}$.

We prove the first case as this one is quite simple, and refer the reader to [Joy21, Determination] for the remaining cases.

Proof of (i). Using Corollary 9.17 the fibrations are determined as $(\mathtt{C} \cap \mathtt{W})^{\boxslash}$. And the cofibrations are determined as ${}^{\boxslash}(\mathtt{F} \cap \mathtt{W})$. So it remains to treat the case where only cofibrations and fibrations are given. We may determine the acyclic cofibrations as ${}^{\boxslash}\mathtt{F}$ and acyclic fibrations as $\mathtt{C}^{\boxslash}$. Every weak equivalence may be factorized as a composite of an acyclic cofibration followed by an acyclic fibration by either factorization system and the 2-out-of-3 property, and conversely, every such composite is a weak equivalence. So $\mathtt{W}$ may be obtained as the class of such composites. □

Definition 10.5 (Model category)**.** A **model category** is a complete and cocomplete category $\mathcal{M}$ with a model structure $(\mathtt{W}, \mathtt{F}, \mathtt{C})$.

Remark 10.6. There is some disagreement among authors as to which completeness criteria are required for model categories. Quillen's original definition [Qui67] postulates only finite completeness and finite cocompleteness, and this is also the definition that Dwyer-Spalinski [DS95] and Joyal [Joy08] uses. We however go with a definition requiring completeness and cocompleteness under arbitrary limits in line with Riehl [Rie09], Riehl–Verity [RV21], and Rasekh [Ras18] as every example we will see satisfies this property.

Remark 10.7. `W`, `F` and `C` are wide subcategories of $\mathcal{M}$ all closed under retracts viewed as objects of $\mathcal{M}^{\mathbb{2}}$. Moreover `F` is stable under products and `C` is stable under coproducts as objects of $\mathcal{M}^{\mathbb{2}}$. Finally, `F` is stable under pullbacks and arbitrary limits of towers while `C` is stable under pushouts and arbitrary colimits of cotowers viewed as objects of $\mathcal{M}$. Because of the completeness properties satisfied by model categories, we are now guaranteed that such limits always exist.

Example 10.8 (Kan–Quillen model structure $\mathbf{sSet}_{\mathrm{KQ}}$, [Qui67])**.** Taking the following classes of morphisms makes **sSet** a model category.

- [`W`] Weak equivalences are morphisms whose geometric realization are weak homotopy equivalences.[19]
- [`C`] Cofibrations are monomorphisms.
- [`F`] Fibrations are Kan fibrations, which can be characterized as having the right lifting property with respect to all horn inclusions

$$\begin{tikzcd} \Lambda^n_k \arrow[r] \arrow[d, tail, "\wr"] & X \arrow[d, two heads, "f"] \\ \Delta^n \arrow[r] \arrow[ur, dashed] & Y. \end{tikzcd}$$

With respect to this model structure, we readily see that all objects are cofibrant and that the Kan complexes are the fibrant objects.

We may also construct a model structure for the category of small categories, having equivalences of categories as its weak equivalences:

Example 10.9 (The model category of categories, [nLabd])**.** The category of small categories **Cat** is a model category with the following classes.

- [`W`] Weak equivalences are the equivalences of categories.
- [`F`] Fibrations are the isofibrations: functors with the right lifting property with respect to the inclusion of either endpoint into the free-standing isomorphism category $\mathbb{I}$.
- [`C`] Cofibrations are the functors that are injective on objects.

[19]There are multiple equivalent characterizations of the weak equivalences, including one which does not rely on any homotopy theory of topological spaces. However, our formulation is the easiest one to state.

Transporting back along the inclusion functor $\mathbf{Gpd} \hookrightarrow \mathbf{Cat}$, this also endows the category of groupoids with the structure of a model category.

Example 10.10 (Other examples of model categories)**.**

(i) Any bicomplete category with isomorphisms as weak equivalences, and all morphisms as both fibrations and cofibrations.

(ii) **Set** with all maps as weak equivalences, surjections as fibrations, and injections as cofibrations.

(iii) **Set** with all maps as weak equivalences, injections as fibrations, and surjections as cofibrations.

(iv) **Top** with weak homotopy equivalences as weak equivalences, Serre fibrations as fibrations, meaning continuous maps with the right lifting property with respect to the inclusions of n-disks $D^n \times \{0\} \hookrightarrow D^n \times I$, and as cofibrations, retracts of relative cell complexes. This is called the Quillen model structure $\mathbf{Top}_{\mathrm{Quillen}}$.

(v) **Top** with homotopy equivalences as weak equivalences, Hurewicz fibrations as fibrations, and closed Hurewicz cofibrations as cofibrations. The Hurewicz fibrations are defined by their right lifting property with respect to all continuous maps $X \cong X \times \{0\} \to X \times I$, and Hurewicz cofibrations are defined by their left lifting property with respect to all maps into the one-point space $X \to \mathbf{1}$. A *closed* Hurewicz cofibration additionally has a closed image. This is called the Strøm model structure $\mathbf{Top}_{\mathrm{Strøm}}$.

10.2 Fibrancy

As already defined, we say an object of a model category $\mathcal{M}$ is fibrant if its unique map to the terminal object is a fibration and that it is cofibrant if the unique map from the initial object is a cofibration. However, using the factorization systems of the model category we may see that every object of a model category is weakly equivalent to both a fibrant and a cofibrant object.

Construction 10.11 ((Co)fibrant replacement)**.** Given any object A in $\mathcal{M}$, we can factorize the unique map $A \to \mathbf{1}$ to obtain an acyclic cofibration to a fibrant object $\mathtt{F}\,A$, called a **fibrant replacement**, and dually we may construct a **cofibrant replacement** by factorizing the unique map $\mathbf{0} \to A$ to obtain an acyclic fibration from a cofibrant object $\mathtt{C}\,A$:

$$\begin{tikzcd}
& \mathtt{C}\,A \arrow[dr, two heads, "\sim"] & & & \\
\mathbf{0} \arrow[ur, tail] \arrow[rr] & & A \arrow[rr] \arrow[dr, tail, "\sim"'] & & \mathbf{1} \\
& & & \mathtt{F}\,A \arrow[ur, two heads] &
\end{tikzcd}$$

Moreover, an arbitrary morphism f lifts to either replacement in a way that preserves commutativity of the following diagram,

$$\begin{tikzcd}
\mathbf{0} \arrow[r, tail] & \mathtt{C}\,A \arrow[r, two heads, "\sim"] \arrow[d, dashed, "\mathtt{C}\,f"] & A \arrow[r, tail, "\sim"] \arrow[d, "f"] & \mathtt{F}\,A \arrow[r, two heads] \arrow[d, dashed, "\mathtt{F}\,f"] & \mathbf{1} \\
\mathbf{0} \arrow[r, tail] & \mathtt{C}\,B \arrow[r, two heads, "\sim"] & B \arrow[r, tail, "\sim"] & \mathtt{F}\,B \arrow[r, two heads] & \mathbf{1},
\end{tikzcd}$$

via the following lifts

$$\begin{tikzcd}
\mathbf{0} \arrow[rr, tail] \arrow[d, tail] & & \mathtt{C}\,B \arrow[d, two heads, "\wr"] \\
\mathtt{C}\,A \arrow[r, two heads, "\sim"] \arrow[urr, dashed, "\mathtt{C}\,f"] & A \arrow[r, "f"] & B
\end{tikzcd}
\quad \text{and} \quad
\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, tail, "\wr"'] & B \arrow[r, tail, "\sim"] & \mathtt{F}\,B \arrow[d, two heads] \\
\mathtt{F}\,A \arrow[rr, two heads] \arrow[urr, dashed, "\mathtt{F}\,f"'] & & \mathbf{1}.
\end{tikzcd}$$

Note that in general these constructions may not be functorial.

By iterated replacement every object may be replaced by a bifibrant one:

$$\begin{tikzcd}
& \mathtt{C}\,A \arrow[dr, two heads, "\wr"] & & \\
\mathbf{0} \arrow[ur, tail] \arrow[rr] & & A \arrow[r] \arrow[dr, tail, "\wr"'] & \mathbf{1} \\
& & & \mathtt{F}\,A \arrow[u, two heads]
\end{tikzcd}
\rightsquigarrow
\begin{tikzcd}
& \mathtt{C}\,A \arrow[rr, tail, "\sim"] \arrow[dr, two heads, "\wr"] \arrow[dd, tail, "\wr"'] & & \mathtt{F}\mathtt{C}\,A \arrow[dr, dashed, two heads] \arrow[dd, two heads, "\wr"] \arrow[dll, dotted, "\wr"] & \\
\mathbf{0} \arrow[ur, tail] \arrow[dr, dashed, tail] & & A \arrow[dr, tail, "\wr"] & & \mathbf{1}. \\
& \mathtt{C}\mathtt{F}\,A \arrow[rr, two heads, "\sim"'] \arrow[uurr, dotted, "\wr"] & & \mathtt{F}\,A \arrow[ur, two heads] &
\end{tikzcd}$$

Both candidates $\mathtt{C}\mathtt{F}\,A$ and $\mathtt{F}\mathtt{C}\,A$ work, and moreover, we have comparison inverse weak equivalences between them by the lifting properties of the inner square of the above-right diagram, depicted as dotted arrows.

10.3 Homotopies

In model categories, we may develop notions of homotopy between maps. For these, we define a notion of path and cylinder objects. These objects generalize path and cylinder objects from topology, by retaining just enough structure to enable a notion of morphism homotopy.

Definition 10.12 (Path and cylinder objects)**.** A **path object** for A is an object $\mathtt{path}\,A$ for which there exists a commutative diagram

$$\begin{tikzcd}
& \mathtt{path}\,A \arrow[dr, "{(d_0,d_1)}"] & \\
A \arrow[ur, "\sim"] \arrow[rr, hook, "{(\mathrm{id},\mathrm{id})}"] & & A \times A.
\end{tikzcd}$$

The path object is **good** if the map $\mathtt{path}\,A \to A \times A$ is a fibration, and **very good** if additionally the weak equivalence $A \xrightarrow{\sim} \mathtt{path}\,A$ is a cofibration.

Dually, a **cylinder object** for A is an object $\mathtt{cyl}\,A$ for which there exists a commutative diagram

$$\begin{tikzcd} A+A \arrow[rr, hook, "(\mathrm{id},\mathrm{id})"] \arrow[dr, "(i_0,i_1)"'] & & A. \\ & \mathtt{cyl}\,A \arrow[ur, "\sim"'] & \end{tikzcd}$$

The cylinder object is **good** if the map $A+A \to \mathtt{cyl}\,A$ is a cofibration, and **very good** if additionally the weak equivalence $\mathtt{cyl}\,A \xrightarrow{\sim} A$ is a fibration.

Note that every object has at least one very good path and cylinder object by the factorization properties of a model structure.

Lemma 10.13. *If A is fibrant and $\mathtt{path}\,A$ is a good path object for A, then the component maps d_0 and d_1 are acyclic fibrations and so, in particular, $\mathtt{path}\,A$ is fibrant.*

Proof. The maps d_i are weak equivalences since we have the factorizations

$$\begin{tikzcd} A \arrow[r, "\sim"] \arrow[rr, bend right, equal] & \mathtt{path}\,A \arrow[r, "d_i"] & A \end{tikzcd}$$

in which two morphisms are already weak equivalences.

Since the projection maps, pr_0 and pr_1 make the following square a pullback

$$\begin{tikzcd} A \times A \arrow[r, "\mathsf{pr}_0"] \arrow[d, "\mathsf{pr}_1"'] \arrow[dr, phantom, "\lrcorner", very near start] & A \arrow[d, two heads] \\ A \arrow[r, two heads] & \mathbf{1} \end{tikzcd}$$

we have by Lemma 9.8 that they are fibrations. Hence since d_i is a composite of fibrations $d_i = \mathsf{pr}_i \circ (d_0 \times d_1)$, it is itself a fibration. □

Definition 10.14 (Right and left homotopy)**.** Two maps $f, g : A \to B$ are said to be **right homotopic**, written $f \sim_r g$, if there exists a path object $\mathtt{path}\,B$ and a map $H : A \to \mathtt{path}\,B$ lifting (f, g)

$$\begin{tikzcd} & \mathtt{path}\,B \arrow[dr, "(d_0,d_1)"] & \\ A \arrow[ur, dashed, "H"] \arrow[rr, "(f,g)"] & & B \times B. \end{tikzcd}$$

The map H is called a right homotopy from f to g via $\mathtt{path}\,B$. Dually, the maps f and g are **left homotopic**, written $f \sim_l g$ if there exists a cylinder object $\mathtt{cyl}\,A$ and a **left homotopy** $H' : \mathtt{cyl}\,A \to B$ extending (f, g)

$$\begin{tikzcd} A+A \arrow[rr, "(f,g)"] \arrow[dr, "(i_0,i_1)"'] & & B. \\ & \mathtt{cyl}\,A \arrow[ur, dashed, "H'"'] & \end{tikzcd}$$

Moreover, the homotopy is said to be good or very good if the path/cylinder object is.

As these notions are dual, every result about one again immediately dualizes to a result about the other. Therefore, we now turn our attention solely to right homotopies and path objects.

First, we establish a series of mechanical results about good and very good homotopies.

Lemma 10.15. *If f and g are right homotopic, then we may construct a good right homotopy.*

Proof. We may construct the following diagram by factorizing d using $(\mathtt{C} \cap \mathtt{W}, \mathtt{F})$ and observing that $\mathtt{path}' B$ is a path object of B

$$\begin{tikzcd}
 & B \arrow[r, dashed, "\sim"] \arrow[d, "\wr"] & \mathtt{path}' B \arrow[dd, two heads, "d'"] \\
 & \mathtt{path}\, B \arrow[ur, tail, "\sim"] \arrow[dr, "d"] & \\
A \arrow[ur, "H"] \arrow[rr, "{(f,g)}"] & & B \times B.
\end{tikzcd}$$

Now, the composition $H' = (A \xrightarrow{H} \mathtt{path}\, B \overset{\sim}{\rightarrowtail} \mathtt{path}' B)$ is our desired good right homotopy.

□

Lemma 10.16. *If f is good right homotopic to g and their domain is cofibrant, then we may construct a very good right homotopy.*

Proof. Let H' be the good right homotopy via $\mathtt{path}' B$. We may factorize the weak equivalence $B \xrightarrow{\sim} \mathtt{path}' B$ via either factorization system and then apply the two-out-of-three property to the other factor to get the square

$$\begin{tikzcd}
B \arrow[r, tail, "\sim"] \arrow[d, "\wr"] & \mathtt{path}'' B \arrow[d, dashed, two heads] \arrow[dl, two heads, "\sim"'] \\
\mathtt{path}' B \arrow[r, two heads, "d'"'] & B \times B
\end{tikzcd}$$

And observe that $\mathtt{path}'' B$ is a very good path object for B. Now if A is cofibrant we may lift the following square

$$\begin{tikzcd}
\mathbf{0} \arrow[r] \arrow[d, tail] & \mathtt{path}'' B \arrow[d, two heads, "\wr"] \\
A \arrow[r, "H'"] \arrow[ur, dashed, "H''"] & \mathtt{path}' B
\end{tikzcd}$$

to obtain our desired very good right homotopy. □

Using these two lemmas, we may in general assume that we have good right homotopies, and when the codomain is fibrant that right homotopies are very good.

Moreover, when our path objects are sufficiently good, we may extend general maps to them in a natural way:

Proposition 10.17 (Extension to path objects)**.** *Assume given objects A and B for which we have respectively a very good path object $\mathtt{path}\,A$, and a good path object $\mathtt{path}\,B$. Then any map $f : A \to B$ extends to a map $\mathtt{path}\,f : \mathtt{path}\,A \to \mathtt{path}\,B$.*

Proof. We obtain $\mathtt{path}\,f$ as the following lift

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, tail, "\wr"'] & B \arrow[r, "\sim"] & \mathtt{path}\,B \arrow[d, two heads] \\
\mathtt{path}\,A \arrow[r, two heads] \arrow[urr, dashed, "\mathtt{path}\,f"] & A \times A \arrow[r, "f \times f"] & B \times B.
\end{tikzcd}$$

∎

Lemma 10.18 (Right homotopy equivalence relation)**.** *If B is fibrant, then right homotopy is an equivalence relation on $A \to B$.*

Proof. See [DS95, Lemma 4.7]. ∎

Lemma 10.19. *If B is fibrant and $p : A' \to A$ is an acyclic fibration, then two maps $f, g : A \to B$ are right homotopic if and only if fp and gp are right homotopic.*

Proof. See [DS95, Lemma 4.9]. ∎

Lemma 10.20 (Right homotopy ideal)**.** *Assume given very good right homotopic maps $f, g : B \to C$ and consider arbitrary maps $p : A \to B$ and $q : C \to D$ and a good path object $\mathtt{path}\,D$. Then we may construct a good right homotopy between qfp and qgp via $\mathtt{path}\,D$.*

Proof. By Proposition 10.17 we have the lift $\mathtt{path}\,q$:

$$\begin{tikzcd}
 & & C \arrow[r, "q"] \arrow[d, tail, "\wr"] & D \arrow[d, "\wr"] \\
A \arrow[r, "p"] & B \arrow[r, "H"] \arrow[dr, "{(f,g)}"'] & \mathtt{path}\,C \arrow[r, "\mathtt{path}\,q"] \arrow[d, two heads] & \mathtt{path}\,D \arrow[d, two heads, "d"] \\
 & & C \times C \arrow[r, "q \times q"] & D \times D.
\end{tikzcd}$$

And the composite $d \circ \mathtt{path}\,q \circ H \circ p$ defines the desired homotopy. ∎

These lemmas ensure that when we transport to fibrant objects, we may quotient right homotopies to get a category $\mathbf{h}\mathcal{M}_{\mathrm{f}}$ whose objects are the fibrant objects of $\mathcal{M}$ and morphisms are right homotopy classes of morphisms in $\mathcal{M}$. And dually for cofibrant objects and left homotopies.

We may now relate these two notions with the following propositions.

Lemma 10.21. *Given maps $f, g : A \to B$ we have the following dual implications.*

(i) If A is cofibrant and $f \sim_l g$ then $f \sim_r g$.

(ii) If B is fibrant and $f \sim_r g$ then $f \sim_l g$.

Proof. See [DS95, Lemma 4.21] or [Rie20, Proposition 3.3.8]. □

Hence when the domain is cofibrant and the codomain is fibrant, the equivalence relations coincide and we simply say that f and g are **homotopic**, denoted $f \sim g$.

Proposition 10.22. *A map f between bifibrant objects A and B has a homotopy inverse if and only if it is a weak equivalence.*

Proof. See [DS95, Lemma 4.24] or [Rie20, Proposition 3.3.10]. □

Using this, we may reframe the problem of localizing a model category in terms of the well-structured homotopies.

Construction 10.23 (Homotopy category of (co)fibrant objects)**.** Given a model category $\mathcal{M}$, we may define the full subcategories of respectively fibrant, cofibrant and bifibrant objects $\mathcal{M}_{\mathrm{f}}$, $\mathcal{M}_{\mathrm{c}}$, and $\mathcal{M}_{\mathrm{cf}}$.

Furthermore, Lemmas 10.18 to 10.20 ensures that for each of these, we may quotient by respectively left homotopy, right homotopy and lastly, by Lemma 10.21, just homotopy to get categories $\mathbf{h}\mathcal{M}_{\mathrm{f}}$, $\mathbf{h}\mathcal{M}_{\mathrm{c}}$, and $\mathbf{h}\mathcal{M}_{\mathrm{cf}}$, and that we may define fibrant replacements functors

$$\mathtt{F} : \mathcal{M} \to \mathbf{h}\mathcal{M}_{\mathrm{f}} \qquad \mathtt{C} : \mathcal{M} \to \mathbf{h}\mathcal{M}_{\mathrm{c}} \qquad \mathtt{CF} : \mathcal{M} \to \mathbf{h}\mathcal{M}_{\mathrm{cf}}.$$

All together we have the following diagram of categories

$$\begin{tikzcd}
 & & \mathcal{M}_{\mathrm{cf}} \arrow[dll, hook'] \arrow[drr, hook] & & \\
\mathcal{M}_{\mathrm{f}} \arrow[d] \arrow[drr, hook] & & & & \mathcal{M}_{\mathrm{c}} \arrow[d] \arrow[dll, hook'] \\
\mathbf{h}\mathcal{M}_{\mathrm{f}} \arrow[dr, "\mathtt{C}"] & & \mathcal{M} \arrow[ll, "\mathtt{F}"'] \arrow[rr, "\mathtt{C}"] & & \mathbf{h}\mathcal{M}_{\mathrm{c}}. \arrow[dl, "\mathtt{F}"'] \\
 & & \mathbf{h}\mathcal{M}_{\mathrm{cf}} & &
\end{tikzcd}$$

Finally, applying the factorization property of the bijective-on-objects, fully faithful orthogonal factorization system as mentioned in Example 9.12(iii) to $\mathtt{CF} : \mathcal{M} \to \mathbf{h}\mathcal{M}_{\mathrm{cf}}$, we get a category we denote as $\mathbf{h}\mathcal{M}$ whose objects are the objects of $\mathcal{M}$, but whose hom-sets consist of homotopy classes of maps between bifibrant replacements

$$\begin{tikzcd}
\mathcal{M} \arrow[r] \arrow[rr, bend right, "\mathtt{CF}"'] & \mathbf{h}\mathcal{M} \arrow[r] & \mathbf{h}\mathcal{M}_{\mathrm{cf}}.
\end{tikzcd}$$

Theorem 10.24
The category $\mathbf{h}\mathcal{M}$ is a localization of $\mathcal{M}$ at $\mathtt{W}$.

Proof. It may be checked directly that $\mathbf{h}\mathcal{M}$ carries the universal property of localizations as stated in Definition 8.7. See for instance [DS95, Theorem 6.2] or [Rie20, Theorem 3.4.5]. □

This construction of the homotopy category of $\mathcal{M}$ is better behaved than the more general calculus of fractions. It is a more tractable construction, and it allows us to utilize the additional structure presented by the model structure. As an example of this, we end this section by remarking on enrichment of model categories.

Remark 10.25 (Enriched model categories, [Rie20, Section 4.4])**.** We may speak of model categories enriched in monoidal model categories. A **monoidal model category** $\mathcal{V}$ is a simultaneous monoidal category and model category that satisfies additional compatibility axioms between these two structures. A **$\mathcal{V}$-enriched model category** is a simultaneous $\mathcal{V}$-enriched category and model category that satisfies appropriate compatibility axioms between these structures. Our essential example of a monoidal model category is the Kan–Quillen model in simplicial sets with the monoidal action given by its cartesian closedness. We call a model category enriched in it a **simplicial model category**.

Part of the power that comes with enriching a model category is that it offers an enrichment of its homotopy category in the homotopy category of its basis of enrichment. For instance, $\mathbf{sSet}_{\mathrm{KQ}}$ is enriched in itself and this gives rise to an enrichment of its homotopy category in Kan complexes.

10.4 Reedy model structure

Given a model category $\mathcal{M}$ and any small index category $\mathcal{J}$, there are two canonical candidates for model structures on the presheaf category $\mathcal{M}^{\mathcal{J}}$ called the **projective** and **injective** model structures respectively.

Definition 10.26 (Projective and injective model structures)**.** Given a model category $\mathcal{M}$ and a small category $\mathcal{J}$. The **projective model structure on $\mathcal{M}^{\mathcal{J}}$** is a model structure (if it exists) whose weak equivalences and fibrations are the pointwise ones. Dually, the **injective model structure on $\mathcal{M}^{\mathcal{J}}$** is a model structure (if it exists) whose weak equivalences and cofibrations are the pointwise ones.

Particularly relevant to us, the category of simplicial sets endowed with the Kan–Quillen model structure enjoys the property of being *combinatorial*[20] which in particular implies that these always exist [HKRS17, Theorem 3.4.1 and Corollary 3.1.7].

However, for our purposes, the consideration of general indexing-categories is not necessary. We care particularly for a specific indexing-category, the simplex category $\mathbf{\Delta}$. This category carries a particularly nice structure which allows us to construct an explicit model structure on $\mathcal{M}^{\mathbf{\Delta}^{\mathrm{op}}}$ for any model category $\mathcal{M}$.

Definition 10.27 (Reedy category)**.** A **Reedy category** is a category $\mathcal{R}$ equipped with a degree map $\deg : \mathrm{Ob}\,\mathcal{R} \to \omega$ for some ordinal ω and two wide subcategories $\mathcal{R}^{+}$ and $\mathcal{R}^{-}$ satisfying the following axioms:

[R+] For any non-identity in $\mathrm{Hom}_{\mathcal{R}^{+}}(a, b)$ we have $\deg a < \deg b$.

[20] A **combinatorial** model category is locally presentable and cofibrantly generated model category. Local presentability is a particular technical property, and we refer the curious reader to [nLabc].

[R–] For any non-identity in $\mathrm{Hom}_{\mathcal{R}^-}(a,b)$ we have $\deg a > \deg b$.

[R-Fac] Every morphism factorizes uniquely as a morphism in $\mathcal{R}^-$ followed by a morphism in $\mathcal{R}^+$.

Proposition 10.28. *The simplex category $\mathbf{\Delta}$ is Reedy with degree map*

$$\deg[n] \coloneqq n \quad : \mathrm{Ob}\,\mathbf{\Delta} \to \mathbb{N}$$

and the Reedy classes $\mathbf{\Delta}^+$, $\mathbf{\Delta}^-$ given by the subcategories of injective and surjective maps respectively.

Proof. The factorization property follows from Lemma 7.3. The rest is clear. □

Some other examples are the following.

Example 10.29 (Examples of Reedy categories)**.** The tautological examples of Reedy categories are the ordinals themselves considered as total order categories. In particular $\mathbb{2}$ and $\mathbb{N}$ regarded as ordered sets are Reedy. Other examples are discrete categories and the diagram categories $(\cdot \rightrightarrows \cdot)$, $(\cdot \leftarrow \cdot \rightarrow \cdot)$ and $(\cdot \rightarrow \cdot \leftarrow \cdot)$.

Definition 10.30 (Latching and matching objects)**.** Given a diagram $X : \mathcal{R} \to \mathcal{M}$, then for any object $r \in \mathrm{Ob}\,\mathcal{R}$ we define the **latching object** $L_r X$ to be the colimit

$$L_r X \coloneqq \operatorname*{colim}_{\substack{(s \to r) \in \mathcal{R}^+/r \\ s \neq r}} X(s).$$

Dually, the **matching object** $M_r X$ is the limit

$$M_r X \coloneqq \lim_{\substack{(r \to s) \in r/\mathcal{R}^- \\ s \neq r}} X(s).$$

When they exist we have a canonical pair of maps

$$L_r X \to X(x) \to M_r X.$$

Example 10.31. In the case of simplicial sets, latching objects can be identified with a collection of degenerate simplices of X, and matching objects can be identified with a collection of faces of X. In particular, $L_{[n]} X \hookrightarrow X[n]$ is the inclusion of degenerate n-simplices.

Theorem 10.32 (Reedy model structure)
Given a Reedy category $\mathcal{R}$ and a model category $\mathcal{M}$, then we may construct a model structure on the presheaf category $\mathcal{M}^{\mathcal{R}}$ for which the weak equivalences are the pointwise ones. A morphism $f : X \to Y$ in $\mathcal{M}^{\mathcal{R}}$ is a cofibration if

$$L_r Y \sqcup_{L_r X} X_r \to Y_r$$

is a cofibration in $\mathcal{M}$ for all $r \in \mathrm{Ob}\,\mathcal{R}$. It is a fibration if

$$X_r \to M_r X \times_{M_r Y} Y_r$$

is a fibration in $\mathcal{M}$ for all $r \in \mathrm{Ob}\,\mathcal{R}$.

Proof. See [RV14, Theorem 4.18]. □

Remark 10.33 ([BR13])**.** The simplex category $\mathbf{\Delta}$ enjoys a special property as a Reedy category in that for every model category $\mathcal{M}$ the Reedy model structure on $\mathcal{M}^{\mathbf{\Delta}}$ coincides with the injective presheaf model structure.

Definition 10.34 (Proper model structure)**.** A model category is said to be **right proper** if weak equivalences are preserved by pullbacks along fibrations. Dually, a model category is **left proper** if weak equivalences are preserved by pushouts along cofibrations. A model category that is both right and left proper is said to be **proper**.

Example 10.35 ([GJ99])**.** The Kan–Quillen model structure on **sSet** is proper.

Proposition 10.36. *Given a Reedy category $\mathcal{R}$ and a left (right) proper model category $\mathcal{M}$, then the Reedy model category $\mathcal{M}^{\mathcal{R}}$ is again left (right) proper.*

Proof. We consider the case of left properness. So assume we have a pushout square

$$\begin{tikzcd} \cdot \arrow[r, tail] \arrow[d, "w"', "\wr"] & \cdot \arrow[d] \\ \cdot \arrow[r] \arrow[ur, phantom, "\ulcorner", very near end] & \cdot \end{tikzcd}$$

in $\mathcal{M}^{\mathcal{R}}$. As a colimit in a functor category taking values in a cocomplete category, this is a pushout if and only if it is a pointwise pushout. Moreover, cofibrations and weak equivalences in $\mathcal{M}^{\mathcal{R}}$ are in particular pointwise cofibrations and weak equivalences. Since $\mathcal{M}$ is left proper, the pushout of w is a weak equivalence at each point. But every pointwise weak equivalence is a weak equivalence in $\mathcal{M}^{\mathcal{R}}$, so we are done. □

Digression 10.37. Our definition of Reedy categories suffers from a certain deficiency. They do not respect the *principle of equivalence*. Given two equivalent categories, there is no way to transport the Reedy structure of one to the other. In fact, a Reedy category cannot have any non-identity isomorphisms. Luckily, this may be redeemed, giving rise to *generalized Reedy categories* as introduced by Clemens Berger and Ieke Moerdijk in [BM10].

11 Models of type theory

Type theories are interesting objects of study by themselves, but their power amplifies when interpreted in some context, enabling us to utilize theorems proven in the type theory to make interesting statements about the context in which it is interpreted. This we call the **semantics** of the type theory, and we call a particular interpretation a **model**. There are in particular many categorical approaches to modeling type theories. See [nLaba] for many such approaches and comparisons between them.

In this section, we take a brief look at how model categories may be used to develop semantics of type theories. Although intensional type theories may be interpreted in less structured categories, model categories nevertheless constitute an important family. In particular, homotopy type theory admits a model in Kan–Quillen simplicial sets [KL18], justifying the interpretation of types as ∞-groupoids and transferring homotopy-theoretic results proven in the type theory to proofs of the corresponding statements in homotopy theory.

It was due to a lack of understanding of the semantics of intensional type theory that the homotopy-theoretic aspects went unnoticed for so long, and through admitting semantics in the category of groupoids [HS98] that mathematicians were first able to refute the principles of the uniqueness of equality proofs and equality reflection for intensional type theory. Neither holds for groupoids, as the first would correspond to only having identity paths and the second would correspond to only having a single object per path component.

Remarkably, it turns out that extensional dependent type theory admits semantics in all locally cartesian closed categories;[21] we say extensional dependent type theory is the *internal language* of locally cartesian closed categories. Correspondingly, intensional type theory with higher inductive types is expected to be the internal language of locally cartesian closed $(\infty,1)$-categories! It had even been conjectured for some time that homotopy type theory with a univalent universe was the internal language of $(\infty,1)$-toposes. This was recently proven by Shulman [Shu19].

What most models have in common (e.g. [HS98; AK11; KL18; RS17]) is the use of a category whose objects are contexts of the type theory under study. A formation of types in context Γ is an object over Γ, i.e. $T \to \Gamma$, and an introduction of terms is a section of this. With this modeling, substitutions are pullbacks

$$\begin{tikzcd} f^{\lrcorner}T \arrow[r, "f^{\lrcorner}"] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & T \arrow[d] \\ \Lambda \arrow[r, "f"] & \Gamma. \end{tikzcd}$$

A non-dependent type is represented by a morphism $A \to ()$, while a dependent type $x : A \vdash B\ \mathsf{type}$ may be represented by a map $B \to A$ whose composability with $A \to ()$ witnesses that we may form the Σ-type $\sum_{x:A} B$. We think of the morphism $B \to A$ as a fibration (this can be made precise), with $B[a/x]$ lying over a:

$$\begin{tikzcd} B[a/x] \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & B \arrow[d] \\ () \arrow[r, "a"] & A. \end{tikzcd}$$

To construct dependent products we require an additional property of the contextual category. Given that all pullbacks along f exist, the *base change functor* $f^{\lrcorner} : \mathcal{C}/\Gamma \to \mathcal{C}/\Lambda$

[21] A locally cartesian closed category is a category $\mathcal{C}$ such that all slice categories $\mathcal{C}/c$ are cartesian closed.

has a left adjoint Σ_f given by composition with f, this forms the dependent sum along f. If $\mathcal{C}$ is moreover locally cartesian closed, $f^{\lrcorner}$ also admits a right-adjoint functor Π_f:

$$\begin{tikzcd} \mathcal{C}/\Gamma \arrow[r, "f^{\lrcorner}"] & \mathcal{C}/\Lambda. \arrow[l, bend right, "\Sigma_f"'] \arrow[l, bend left, "\Pi_f"] \arrow[l, phantom, shift right=3, "\perp"] \arrow[l, phantom, shift left=3, "\perp"] \end{tikzcd}$$

This right-adjoint to the base change functor is precisely the categorical formulation of dependent products along f and corresponds to the formation of Π-types along f. For instance, in the case of a dependent type $x : A \vdash B$ type as considered above, a dependent product along $A \to ()$ yields $\prod_{x:A} B$. This adjointness relation between existential quantification, substitution, and universal quantification was first remarked by Lawvere in [Law06].

Groupoid model. Hofmann–Streicher interpreted Martin–Löf dependent type theory with a universe type in the category of groupoids [HS98]. In this model, types are interpreted as groupoids and open terms, meaning terms judged in a non-empty context, are interpreted as functors between groupoids.[22] The functoriality of these maps represents the preservation of identity terms. Dependent types require a notion of type families, which conveniently corresponds to functors $P : A \to \mathbf{Gpd}$. In particular, since all morphisms in A are isomorphisms, fibers over an arrow a are isomorphic, witnessed by the functorial action of P on a. Note that an internal universe must correspond to a *groupoid* of small groupoids.

∞-groupoid model. Homotopy type theory with a univalent universe admits a model in the Kan–Quillen model of simplicial sets [KL18; Str14] usually attributed to Voevodsky. In this model, type families are fibrations, and so types correspond to Kan complexes. This solidifies our understanding of types as ∞-groupoids, in addition to giving semantics to the synthetic homotopy theory of the type theory. Through this model, a formalized proof of some homotopy-theoretic result is also a proof of the homotopy-theoretic result in the classical setting!

Digression 11.1 (The constructive Kan–Quillen model structure)**.** There is a constructive model structure on simplicial sets which under the assumption of the law of excluded middle and axiom of choice coincides with the Kan–Quillen model structure, presented in the paper [GSS19]. In the context of modeling intensional type theory, a constructive model structure is desirable as it allows for internalization of the model, and hence, in particular, reflection on it.

Complications arise in the constructive case. First of all, not all objects are cofibrant. Luckily, this may be circumvented with a strong cofibrant replacement. Moreover, a distinction is made between *trivial* and *acyclic* fibrations. The first is defined as having

[22] Closed terms are also functors, but only trivially as projections to the trivial groupoid.

the right lifting property with respect to all cofibrations, while the second is defined to mean it is a weak equivalence and a fibration. Part of their work includes showing that these two classes coincide, a non-trivial fact in the constructive setting.

There are some results on the classical formulation that simply aren't provable constructively, however. In [BCP15], they for instance show that this is the case for the result that given two Kan complexes X and Y, their exponential Y^X is again a Kan complex.

12 Simplicial spaces

We now turn our attention to the category of simplicial spaces inside of which we obtain **Segal spaces** and **Rezk spaces** also called **complete Segal spaces**. These are respectively precategory and category objects in the category of Kan complexes and model $(\infty, 1)$-categories. These objects are of particular interest to us as they form the underlying objects in the motivating model of the simplicial type theory we present in the next part.

We will apply the language of model categories to efficiently present their homotopy theory. We start with the homotopy theory of Kan–Quillen simplicial sets and study their simplicial objects. These are called **simplicial spaces** or **bisimplicial sets** and come equipped with the Reedy model structure presenting the homotopy theory of simplicial ∞-groupoids. Hence they are objects with a spatial and a simplicial (later categorical) direction. Assuming the Segal condition in the simplicial direction leads us to the study of **Segal spaces**, which act like categories in the simplicial direction. These also admit a homotopy theory presented by a model structure that may concisely be defined as the left Bousfield localization of the Reedy model structure at the spine inclusions. With Segal spaces we encounter the same issues as displayed by precategories in homotopy type theory: The categorical structure of a Segal space does not sufficiently interact with its homotopical structure to model $(\infty, 1)$-categories. This we may remedy by adding a condition of local univalence, which leads us to the definition of Rezk spaces. Their homotopy theory may again be presented by a model structure defined as the left Bousfield localization of the Segal model structure at the terminal projection of (the discrete nerve of) the free-standing isomorphism.

We focus on definitions and examples, skimming over technical details and leaving out proofs entirely. We will treat the homotopy and category theory of these objects in more detail in the context of the type theory they interpret, which is the subject of the next part.

Remark 12.1. There are many models presenting the homotopy theory of $(\infty, 1)$-categories. We have model categories themselves, presenting bicomplete $(\infty, 1)$-categories. Since general $(\infty, 1)$-categories need not be bicomplete, model categories are not sufficient for the presentation of all $(\infty, 1)$-categories. For this, we can take multiple, appropriately equivalent, approaches.

The subject of this section is one such model, namely the theory of Rezk spaces. Another model is the model of simplicially enriched categories. Indeed, we have already seen an instance of this model with the hammock localization of homotopical categories as outlined in Remark 8.11.

The best-established model of $(\infty,1)$-categories, however, is the theory of quasi-categories as defined by Boardman and Vogt. These admit a rather simple description as simplicial sets satisfying a simultaneous relaxation of the (inner horn formulation of the) Segal condition and the Kan condition: all inner horns have fillers. Hence quasi-categories are also referred to as inner Kan complexes. Their treatment is much due to Joyal [Joy08] and Lurie [Lur09]. In fact, quasi-categories even form the basis of enrichment for the model-independent approach to $(\infty,1)$-category theory taken by Riehl–Verity [RV21].

Definition 12.2. A **simplicial space** or **bisimplicial set** is a **sSet**-valued presheaf on the simplex category, meaning it is a functor

$$\mathbf{\Delta}^{\mathrm{op}} \to \mathbf{sSet},$$

and we define the category of simplicial spaces to be $\mathbf{ssSet} \coloneqq \mathbf{sSet}^{\mathbf{\Delta}^{\mathrm{op}}}$.

Some authors take simplicial spaces to more generally mean preshaves on the simplex category taking values in *any* category modeling the homotopy theory of topological spaces and so the terminology **bisimplicial set**, using the product-hom adjunction

$$\left(\mathbf{Set}^{\mathbf{\Delta}^{\mathrm{op}}}\right)^{\mathbf{\Delta}^{\mathrm{op}}} \cong \mathbf{Set}^{\mathbf{\Delta}^{\mathrm{op}} \times \mathbf{\Delta}^{\mathrm{op}}},$$

may be considered more specific. We will however not consider any other such model.

From the viewpoint of simplicial spaces as bisimplicial sets we may visualize a simplicial space X as a grid of sets related by face and degeneracy maps in two independent directions. The first argument varies vertically, and the second argument varies horizontally:

$$\begin{array}{ccccccc}
\vdots & & \vdots & & \vdots & & \iddots \\
\downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow & & \downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow & & \downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow & & \\
X_{2,0} & \substack{\longleftarrow \\ \longrightarrow \\ \longleftarrow} & X_{2,1} & \substack{\longleftarrow \\ \longrightarrow \\ \longleftarrow \\ \longrightarrow \\ \longleftarrow} & X_{2,2} & \substack{\longleftarrow \\ \longrightarrow \\ \longleftarrow \\ \longrightarrow \\ \longleftarrow \\ \longrightarrow \\ \longleftarrow} & \cdots \\
\downarrow\uparrow\downarrow\uparrow\downarrow & & \downarrow\uparrow\downarrow\uparrow\downarrow & & \downarrow\uparrow\downarrow\uparrow\downarrow & & \\
X_{1,0} & \substack{\longleftarrow \\ \longrightarrow \\ \longleftarrow} & X_{1,1} & \substack{\longleftarrow \\ \longrightarrow \\ \longleftarrow \\ \longrightarrow \\ \longleftarrow} & X_{1,2} & \substack{\longleftarrow \\ \longrightarrow \\ \longleftarrow \\ \longrightarrow \\ \longleftarrow \\ \longrightarrow \\ \longleftarrow} & \cdots \\
\downarrow\uparrow\downarrow & & \downarrow\uparrow\downarrow & & \downarrow\uparrow\downarrow & & \\
X_{0,0} & \substack{\longleftarrow \\ \longrightarrow \\ \longleftarrow} & X_{0,1} & \substack{\longleftarrow \\ \longrightarrow \\ \longleftarrow \\ \longrightarrow \\ \longleftarrow} & X_{0,2} & \substack{\longleftarrow \\ \longrightarrow \\ \longleftarrow \\ \longrightarrow \\ \longleftarrow \\ \longrightarrow \\ \longleftarrow} & \cdots
\end{array}$$

where we take $X_{n,m}$ to correspond to $(X_n)_m$ under the product-hom adjunction. As a natural construction we may build a bisimplicial set from two simplicial sets in this way:

Definition 12.3 (External product)**.** We have a bifunctor from simplicial sets to bisimplicial sets defined on objects as

$$- \boxtimes - : \mathbf{sSet} \times \mathbf{sSet} \to \mathbf{ssSet} \qquad (A \boxtimes B)_{n,m} := A_n \times B_m$$

and with the obvious action on maps.

In particular, we have the represented functors

$$\Delta^n \boxtimes \Delta^m = \mathrm{Hom}_{\mathbf{\Delta}\times\mathbf{\Delta}}(-, ([n], [m])).$$

We also have a horizontal embedding of simplicial sets into bisimplicial sets given by $\Delta^0 \boxtimes (-)$, and a vertical embedding $(-) \boxtimes \Delta^0$.

Just as **sSet**, the category **ssSet** enjoys many nice categorical properties.

Proposition 12.4. *The category* **ssSet** *is bicomplete and cartesian closed.* □

And we have explicit characterizations of products and internal homs:

Definition 12.5 (Product and internal hom)**.** Given simplicial spaces X and Y, we may define $X \times Y$ and Y^X degreewise as

$$(X \times Y)_{n,m} = X_{n,m} \times Y_{n,m} \qquad \text{and} \qquad (Y^X)_{n,m} = \mathrm{Hom}_{\mathbf{ssSet}}((\Delta^n \boxtimes \Delta^m) \times X, Y).$$

In particular we consider **ssSet** as enriched in simplicial sets by projecting the simplicial space Y^X onto its 0'th horizontal complex $(Y^X)_0 = (Y^X)_{0,\bullet}$.

Proposition 12.6 (Reedy model structure on simplicial spaces $\mathbf{sSet}_{\mathrm{Reedy}}$)**.** *Using the Reedy model structure on* $\mathbf{\Delta}$ *and the Kan–Quillen model structure on simplicial sets* $\mathbf{sSet}_{\mathrm{KQ}}$*, we endow the category of simplicial spaces with a Reedy model structure* $\mathbf{ssSet}_{\mathrm{Reedy}}$*. Its weak equivalences are hence pointwise weak equivalences, cofibrations are just monomorphisms since the model structure is injective, and fibrations are maps* $f : X \to Y$ *such that for all* $n \in \mathbb{N}$ *the induced dashed arrow*

$$\begin{tikzcd}
X^{\Delta^n\boxtimes\Delta^0} \arrow[rr] \arrow[dr, dashed] \arrow[dd, "f^{\Delta^n\boxtimes\Delta^0}"'] & & X^{\partial\Delta^n\boxtimes\Delta^0} \arrow[dd, "f^{\partial\Delta^n\boxtimes\Delta^0}"] \\
& \cdot \arrow[ur] \arrow[dl] \arrow[dr, phantom, "\lrcorner", very near start] & \\
Y^{\Delta^n\boxtimes\Delta^0} \arrow[rr] & & Y^{\partial\Delta^n\boxtimes\Delta^0},
\end{tikzcd}$$

called the Leibniz cotensor of the inclusion $\partial\Delta^n \boxtimes \Delta^0 \hookrightarrow \Delta^n \boxtimes \Delta^0$ *with* f*, is a Kan fibration.*

All simplicial spaces are thus cofibrant, and the fibrant objects are simplicial spaces X *such that each restriction map*

$$\left(X^{\Delta^n\boxtimes\Delta^0}\right)_0 \to \left(X^{\partial\Delta^n\boxtimes\Delta^0}\right)_0$$

is a fibration of spaces. In particular, this implies that each $X_{n,\bullet}$ *is a Kan complex.*

Since the underlying model category is proper this model structure is also proper.

With the Reedy model structure, we recover a homotopy theory of spaces in the horizontal direction, hence the Reedy model structure gives a homotopy theory of simplicial ∞-groupoids.

12.1 Segal spaces

Definition 12.7 (Segal space)**.** A Segal space is a Reedy fibrant simplicial space X satisfying the Segal condition in the categorical direction. By which we mean that the canonical maps

$$\left(X^{\Delta^n \boxtimes \Delta^0}\right)_0 \to \left(X^{\mathrm{Sp}^n \boxtimes \Delta^0}\right)_0$$

are Kan equivalences. We say X is **local** with respect to the inclusions

$$\mathrm{Sp}^n \boxtimes \Delta^0 \hookrightarrow \Delta^n \boxtimes \Delta^0.$$

Using the techniques of Bousfield localization, we may construct a model structure presenting the homotopy theory of Segal spaces. The class of cofibrations is kept fixed, while the class of fibrations is adjusted to accommodate an enlarged class of weak equivalences.

Theorem 12.8 (Model of Segal spaces $\mathbf{ssSet}_{\mathrm{Segal}}$, [Rez01, Theorem 7.1])
Segal spaces form the fibrant objects of the left Bousfield localization of $\mathbf{ssSet}_{\mathrm{Reedy}}$ *at the inclusions*

$$\mathrm{Sp}^n \boxtimes \Delta^0 \hookrightarrow \Delta^n \boxtimes \Delta^0.$$

We may characterize its model structure classes as follows.

- [C] *The cofibrations are the monomorphisms.*
- [F] *The fibrant objects are the Segal spaces, and a map between Segal spaces is a Segal fibration if and only if it is a Reedy fibration.*
- [W] *Weak equivalences are the maps f such that* $\mathrm{Hom}_{\mathbf{ssSet}}(f, X)_0$ *is a Kan equivalence for every Segal space X. I.e. weak equivalences are the maps which are seen to induce equivalences of mapping spaces by Segal spaces. We note that Reedy equivalences are Segal equivalences, and that Segal equivalences between Segal spaces are Reedy equivalences.*

Moreover, this model structure is cofibrantly generated, left proper, and may be enriched in $\mathbf{sSet}_{\mathrm{KQ}}$.

With the Segal space model, we get a homotopy theory in the horizontal direction and a category theory in the vertical direction.

Example 12.9. The simplicial spaces $\Delta^n \boxtimes \Delta^0$, $\mathcal{N}(\mathbb{I}) \boxtimes \Delta^0$ and more generally the *discrete nerves* $\mathcal{N}(\mathcal{C}) \boxtimes \Delta^0$ for any small category $\mathcal{C}$ are Segal spaces.

The following example demonstrates that the model of Segal spaces is not right proper.

Example 12.10 (Counterexample to right properness)**.** We have the pullback square

$$\begin{tikzcd} \partial\Delta^1 \boxtimes \Delta^0 \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & \Delta^1 \boxtimes \Delta^0 \arrow[d, "d_1 \boxtimes \mathrm{id}"] \\ \mathrm{Sp}^2 \boxtimes \Delta^0 \arrow[r, hook, "\sim"] & \Delta^2 \boxtimes \Delta^0 \end{tikzcd}$$

The right-hand vertical map is a Segal fibration since it is a Reedy fibration between Segal spaces. Moreover, the bottom inclusion is a Segal equivalence by definition. However, its pullback is not a Segal equivalence, as $\partial\Delta^1 \boxtimes \Delta^0$ is not Segal equivalent to $\Delta^1 \boxtimes \Delta^0$.

12.2 Rezk spaces

Segal spaces have a category theory and a homotopy theory, however, these do not sufficiently interact to model $(\infty, 1)$-categories. To demonstrate this, consider the following example:

Example 12.11. The category $\mathbb{I}$ is equivalent to $\mathbf{1}$, and the simplicial set $\mathcal{N}(\mathbb{I})$ is Kan equivalent to Δ^0. However, the Segal space $\mathcal{N}(\mathbb{I}) \boxtimes \Delta^0$ is not Segal equivalent to the terminal object $\mathbf{1}$. In the homotopical direction, we observe that $(\mathcal{N}(\mathbb{I}) \boxtimes \Delta^0)_{0,m} = \mathcal{N}(\mathbb{I})_0 \times \Delta^0_m = \{0, 1\} \times \{s_0^m(0)\}$ which has cardinality 2: homotopically the simplicial space is seen as having two unrelated vertices although categorically they are equivalent.

We expect $(\infty, 1)$-categories to satisfy an additional completeness criterion in this regard: that every isomorphism also reflects to a path. We expect the homotopy theory in $(\infty, 1)$-categories to also capture the homotopy theory in 1-categories. This leads us to the definition of Rezk spaces.

Definition 12.12 (Rezk space)**.** A Rezk space is a Segal space which is local with respect to the terminal projection $\mathcal{N}(\mathbb{I}) \boxtimes \Delta^0 \to \mathbf{1}$. I.e. Rezk spaces see this projection as an equivalence of mapping spaces.

Theorem 12.13 (Model of Rezk spaces $\mathbf{ssSet}_{\mathrm{Rezk}}$, [Rez01, Theorem 7.2])
Rezk spaces form the fibrant objects of the left Bousfield localization of $\mathbf{ssSet}_{\mathrm{Segal}}$ *at the terminal projection* $\mathcal{N}(\mathbb{I}) \boxtimes \Delta^0 \to \mathbf{1}$.

- [C] *The cofibrations are the monomorphisms.*
- [F] *The fibrant objects are precisely the Rezk spaces, and a map between Rezk spaces is a Rezk fibration if and only if it is a Reedy fibration.*
- [W] *Weak equivalences are the maps* f *such that* $\mathrm{Hom}_{\mathbf{ssSet}}(f, X)_0$ *is a Kan equivalence of spaces for every Rezk space* X. *We note that Reedy equivalences are Rezk equivalences, and that Rezk equivalences between Rezk spaces are Reedy equivalences.*

Moreover this model structure is cofibrantly generated, left proper and may be enriched in $\mathbf{sSet}_{\mathrm{KQ}}$.

Considering the failure of the discrete nerve of categories to be a Rezk space, the following alternative construction is proposed.

Example 12.14 (Classification diagram)**.** Given a small category $\mathcal{C}$, then we may define its **classification diagram** $\mathfrak{N}(\mathcal{C})$ as a simplicial space such that

$$\mathfrak{N}(\mathcal{C})_{n,m} \coloneqq \mathrm{Hom}_{\mathbf{Cat}}([n] \times \mathbb{I}(m), \mathcal{C})$$

where $\mathbb{I}(m)$ is the indiscrete category on m objects, which we may define to be the groupoidal localization $\mathrm{Loc}[m]$ in which case we immediately have induced face and degeneracy maps.

This simplicial space is a Rezk space [Rez01]. As the description makes apparent, the categorical direction reflects the categorical structure of $\mathcal{C}$ directly, while the homotopical structure restricts to the core of $\mathcal{C}$. In particular, we recover $\mathcal{N}(\mathrm{Core}\,\mathcal{C})$ as $\mathfrak{N}(\mathcal{C})_0$.

Part III
Simplicial Homotopy Type Theory

Homotopy type theory, whose objects form ∞-groupoids via identity proofs and higher inductive types, may be understood as a theory of homotopy types. The utility of the homotopy invariance of all internal constructions in this type theory can not be exaggerated. We would like to extend this utility to the context of higher categories. However, it is an open problem to formulate the notion of a $(\infty, 1)$-category internally to homotopy type theory, although many attempts are being made [Kra21]. The problem seems to boil down to the fact that all known formulations of $(\infty, 1)$-categories require an infinite amount of coherence axioms.

Although there does not seem to be any intuition among mathematicians for why such a construction would be impossible, a solution remains elusive. Moreover, it may happen that even if such a construction is possible, that it is just too complicated to be practical in application. As an alternative approach, we may extend homotopy type theory in a multitude of ways to enable reasoning about these objects. One approach adds a notion of directed shapes as a basis for the geometry of $(\infty, 1)$-categories. This leads us to the simplicial homotopy type theory as presented by Riehl–Shulman in [RS17], which extends Martin–Löf dependent type theory with additional layers supporting a fundamental theory of strict directed shapes.

This type theory expresses a synthetic theory of simplicial ∞-groupoids, motivated by the Reedy model structure on simplicial spaces. In it, we recover a notion of $(\infty, 1)$-precategories, named Segal types due to their correspondence with Segal spaces in the motivating model. These objects are characterized internally in terms of one singular axiom; namely that compositions of arrows exist and are homotopy unique. Due to this axiom being formulated internally to the theory, it is stronger than it may seem at first glance. It in fact formulates that the restriction map $B^{\Delta^2} \to B^{\Lambda^2_1}$ is *an equivalence of simplicial spaces*. Moreover, when specializing to Segal spaces all internal constructions are automatically functorial due to the simplicial underpinnings.

We further recover $(\infty, 1)$-categories, named Rezk types due to their correspondence with Rezk spaces in the motivating model. These are Segal types satisfying the local univalence condition: the type of isomorphisms between terms are equivalent to the type of homotopies between them. Notably, for the reader familiar with the theory of Rezk spaces, we will see that Segal types without a local univalence condition are sufficient for many $(\infty, 1)$-categorical results. In fact, many of the "categorical" constructions we discuss have meaning for arbitrary types in the theory.

The homotopy invariance of constructions makes arguments so pain-free in this type theory that one could even argue that working with ∞-categories homotopy invariantly is *easier* than working with 1-category theory in classical set theory. Although of course, it limits us from using some of the expressive tools of classical mathematics, it ensures the absence of "evil" constructions and enables us to make efficient arguments without transporting to homotopy-truncated objects like Riehl and Verity do in their model-independent approach to $(\infty, 1)$-category theory [RV21] from which we take much inspiration.

The type theory is still in early development, with the first paper published in 2017 by Emily Riehl and Mike Shulman [RS17]. However, it has gathered some interest and investigations are being done into among others a directed analog of univalence by Emily Riehl, Evan Cavallo and Christian Sattler [RCS18], a cubical formulation of this by Matthew Weaver and Dan Licata [WL20], and Ulrik Buchholtz and Jonathan Weinberger's development of the synthetic theory of fibered $(\infty, 1)$-category theory in [BW21].

We start in Section 13 with defining the simplicial type theory layer by layer. We present it formally as a deductive system like in Section 2. Then in Section 14 we investigate the simplicial nature of types as well as the properties of relative extension types. In Section 15 we define Segal and Rezk types, and establish their basic category and homotopy theory, before looking at a few categorical constructions. Finally, in Section 16 we discuss the notions of families, fibrations, and lifting properties, working towards cocartesian fibrations and families. Most notably, we formulate the notion of absolute lifting diagrams and establish basic properties for these. In particular, we prove a comma representability theorem, which can be used to give a new internal characterization of cocartesian fibrations. This internal characterization was suggested by Emily Riehl, and we base our work mostly on work done by Buchholtz–Weinberger [BW21] and Riehl–Verity [RV21].

Cocartesian families can concisely be summarized as functorial type families with categorical fibers. Like the Grothendieck construction of Theorem 6.9, we have straightened and unstraightened equivalents of cocartesian fibrations, hence they are tools allowing us to switch between these two viewpoints. In particular, there is an internal Yoneda lemma for (co)cartesian fibrations[BW21, Section 7] which can be seen as a principle of directed arrow induction for functorial families of Rezk types.

13 Defining simplicial type theory

The type theory is implemented using multiple layers, meaning we have multiple classes of type-like objects, each class building on the previous one. In this simplicial type theory, the bottom two levels express an intuitionistic first-order logic of strict directed shapes. There is a sense in which the axioms given are precisely the axioms needed to produce an intuitionistic theory of strict directed shapes. On top of this layer, we place a bare-bones Martin–Löf dependent type theory without an internal universe as defined in Section 3.1. It may be remarked that on this layer, every result from Part I still applies. From shapes to types we assert a type of relative functions and assume an axiom of relative function extensionality. These relative functions express shape inclusions as cofibrations. In particular, we may formulate a homotopy extension property for them internally.

This fact is crucial to the homotopy invariance of the type theory. In the presence of strict equality, which is semantically interpreted as the (not generally a fibration) diagonal map $A \to A \times A$, care has to be taken to ensure that all types remain fibrant so that all type constructions stay homotopy invariant.

The type theory as defined by Riehl and Shulman does not assume the existence of a universe type nor univalence. This is not due to incompatibility with these axioms, but as these are needed for the results the authors present, this allows them to maintain a higher level of generality. In our second source [BW21], Buchholtz and Weinberger *do* assume a universe type as well as univalence. Moreover, they coerce the directed shapes to also be types, allowing one, in particular, to construct maps from types into shapes internally.

We will avoid assuming the existence of a universe type in the first half, where we discuss the basic type theory and category theory. However, in the second half, we turn our attention to ideas of fibrations as previously treated in the context of homotopy type theory in Section 6.2. In this context, we introduce a univalent universe again, in particular, to make use of the straightening-unstraightening constructions. It may be noted that the universe type is not a Rezk or Segal type itself in our type theory, but merely a simplicial ∞-groupoid. The coercion of shapes to types on the other hand seems to be more of a convenience than a necessity, as shape inclusions are still definable syntactically which is sufficient for our purposes. Hence we do not assume this kind of coercion.

We now begin the presentation of the type theory, building on the syntax as presented in Section 2.

Context. A context is as before a finite list of variables. However, now variables will be different things at each level of the type theory. In the cube-layer, we have cube-variables, hypothesized terms of finite-dimensional directed cubes. In the next layer up, we find tope-variables which are hypotheses that a formula expressed by a tope is satisfied. For instance, it is valid to hypothesize that $\bot$, the trivial falsity, holds though this is never true. Another way of stating this is that $(\bot)$ is a well-formed context. In general, a variable is an assumption that some well-formed object is inhabited. We use the syntax $\Gamma\ \mathsf{ctx}$ for the judgment that Γ is a well-formed context. The notation Γ, Λ is taken to mean the juxtaposition of Γ followed by Λ, where Γ and Λ are well-formed contexts and Λ may depend on Γ. If the contexts Γ and Λ live in different layers of a multi-level type theory we may separate them by a vertical bar instead. We will usually assume ctx-judgments implicitly, as they should be self-evident from the environment. Given a variable-judgment γ we will say that Γ *contains* γ, denoted $\gamma \in \Gamma$, when Γ is of the form $\Gamma', \gamma, \Gamma''$.

Notational quantification. We will be using a notation for implicit quantification in our inference rules with the intension of making them somewhat more readable. We introduce indexing variables i, j which we implicitly quantify over. Subscript-indexing means universal quantification and parenthesis-indexing meaning existential quantification as long as we're not quantifying over anything that could make this ambiguous (e.g.

function terms). For instance

$$\frac{\Gamma \vdash \phi_i}{\Gamma \vdash \phi_1 \wedge \phi_2} \quad \text{means} \quad \frac{\Gamma \vdash \phi_1 \qquad \Gamma \vdash \phi_2}{\Gamma \vdash \phi_1 \wedge \phi_2},$$

$$\text{while} \quad \frac{\Gamma \vdash \phi(i)}{\Gamma \vdash \phi(1) \vee \phi(2)} \quad \text{means} \quad \frac{\Gamma \vdash \phi(1)}{\Gamma \vdash \phi(1) \vee \phi(2)} \quad \text{and} \quad \frac{\Gamma \vdash \phi(2)}{\Gamma \vdash \phi(1) \vee \phi(2)}.$$

13.1 Cubes

The bottom layer expresses a non-dependent intuitionistic type theory of finite products of the directed interval $\mathbb{2}$. We achieve this by stipulating two basic cubes, the directed interval itself, and a one-term cube $\mathbf{1}$ playing the role of the empty product. We then define a binary product operation with the expected formation, introduction, and elimination rules. Hence a general cube is (up to reparenthesization) just a finite power of the directed interval $\mathbb{2}$.

$$\frac{I\ \mathsf{cube}}{(t : I)\ \mathsf{ctx}}(\text{cube-exten}) \qquad \frac{(t : I) \in \Xi}{\Xi \vdash t : I}(\text{cube-var}) \qquad \frac{}{() \vdash \mathbf{1}\ \mathsf{cube}}(\mathbf{1}\text{-form})$$

$$\frac{}{\Xi \vdash \langle\rangle : \mathbf{1}}(\mathbf{1}\text{-intro})$$

$$\frac{}{() \vdash \mathbb{2}\ \mathsf{cube}}(\mathbb{2}\text{-form}) \qquad \frac{}{\Xi \vdash 0 : \mathbb{2} \qquad \Xi \vdash 1 : \mathbb{2}}(\mathbb{2}\text{-intro})$$

$$\frac{() \vdash I_i\ \mathsf{cube}}{() \vdash I_1 \times I_2\ \mathsf{cube}}(\times\text{-form}) \qquad \frac{\Xi \vdash t_i : I_i}{\Xi \vdash \langle t_1, t_2\rangle : I_1 \times I_2}(\times\text{-intro}) \qquad \frac{\Xi \vdash t : I_1 \times I_2}{\Xi \vdash \mathsf{pr}_i\, t : I_i}(\times\text{-elim})$$

Figure 10: The cube layer

We will continue to use Ξ to mean a well-formed context of cube-variables, which we call a cube context. These rules comprise all the inference rules on the cube layer. Hence we may conclude that a general cube is a finite power (including 0) of $\mathbb{2}$

$$I = \mathbb{2}^m$$

and a general cube context is a finite list of cube-variables

$$\Xi \ = \ (t_1 : \mathbb{2}^{i_1}, \dots, t_n : \mathbb{2}^{i_n}).$$

Of course, the cube $(\mathbb{2} \times \mathbb{2}) \times \mathbb{2}$ is not judgmentally equal to the cube $\mathbb{2} \times (\mathbb{2} \times \mathbb{2})$, hence the notation $\mathbb{2}^3$ is ambiguous. However, just as one would expect, there are meta-theoretic inverse maps between them which essentially reparenthesises the expressions

$$t \mapsto \big\langle \mathsf{pr}_0^2\, t, \langle \mathsf{pr}_0\, \mathsf{pr}_1\, t, \mathsf{pr}_1\, t\rangle \big\rangle \quad \text{and} \quad s \mapsto \big\langle \langle \mathsf{pr}_0\, s, \mathsf{pr}_1\, \mathsf{pr}_0\, s\rangle, \mathsf{pr}_1^2\, s \big\rangle.$$

The same goes for all parenthesizations of $\mathbb{2}^n$ for any n. For definiteness, when we refer to a cube $\mathbb{2}^n$ without relation to another cube, we will mean the right-associated cube $\mathbb{2} \times (\mathbb{2} \times (\cdots \times \mathbb{2}) \dots)$. However, in accordance with [RS17] we will continue to be indifferent to the exact parenthesizations when making statements like $\mathbb{2}^n \times \mathbb{2}^m = \mathbb{2}^{n+m}$ and will write a general term of either without internal parentheses, $\langle t_1, \dots, t_{n+m} \rangle$, as this will have no effect on our arguments.

Remark 13.1. Since our bottom layer never references judgments from the above layers, cubes and cube-variables may never depend on tope or type variables. So we may always place cube-variables before all others in a context. The same also goes for variables of the second and top layers. Hence we may always decompose a general context into a cube context followed by a tope context followed by a type context

$$\Xi \mid \Phi \mid \Gamma.$$

13.2 Topes

On top of the simple intuitionistic theory of finite cubes, we place an intuitionistic first-order logic of **topes**, which we interpret as polytopes living inside cubes. We have four basic topes, the empty tope $\bot$, the full tope $\top$, the strict equality tope $\equiv$ expressing the bottom layer as an intuitionistic theory of *strict* cubes. Note that although this uses the same symbol as judgmental equality, they are different things. Finally, we have the inequality tope $\leq$ expressing the generating cube $\mathbb{2}$ as a directed interval.

To take an example, the tope $s \leq t$ inside the directed 2-cube $\langle t, s \rangle : \mathbb{2} \times \mathbb{2}$ is interpreted as the filled directed lower triangle

$$\begin{tikzcd}
\langle 0,1 \rangle \arrow[r, dotted, "s\equiv 1"] & \langle 1,1 \rangle \\
\langle 0,0 \rangle \arrow[u, dotted, "t\equiv 0"] \arrow[ur, "s\equiv t"] \arrow[r, "s\equiv 0"] & \langle 1,0 \rangle . \arrow[u, "t\equiv 1"']
\end{tikzcd}$$

In addition to the basic topes, we have two basic operations on topes, intuitionistic disjunction[23] $\vee$ and conjunction $\wedge$. We do not postulate operations of negation, implication, existential quantification, or universal quantification. Note in particular that an operation of negation in conjunction with the inequality tope would give us an operation of path reversal, which we do not want as this would make all arrows invertible.

[23]We say the disjunction is *intuitionistic* because a proof of this kind of disjunction contains enough information to recover which case was true to begin with.

$$\frac{\Xi \vdash \phi\ \mathsf{tope}}{\Xi, \phi\ \mathsf{ctx}}(\mathsf{tope}\text{-exten}) \qquad \frac{\phi \in \Phi}{\Xi \mid \Phi \vdash \phi}(\mathsf{tope}\text{-var})$$

$$\frac{}{\Xi \vdash \bot\ \mathsf{tope}}(\bot\text{-form}) \qquad \frac{\Xi \mid \Phi \vdash \bot}{\Xi \mid \Phi \vdash \phi}(\bot\text{-elim}) \qquad \frac{}{\Xi \vdash \top\ \mathsf{tope}}(\top\text{-form}) \qquad \frac{}{\Xi \mid \Phi \vdash \top}(\top\text{-intro})$$

$$\frac{\Xi \vdash \phi_i\ \mathsf{tope}}{\Xi \vdash (\phi_1 \wedge \phi_2)\ \mathsf{tope}}(\wedge\text{-form}) \qquad \frac{\Xi \mid \Phi \vdash \phi_i}{\Xi \mid \Phi \vdash \phi_1 \wedge \phi_2}(\wedge\text{-intro}) \qquad \frac{\Xi \mid \Phi \vdash \phi_1 \wedge \phi_2}{\Xi \mid \Phi \vdash \phi_i}(\wedge\text{-elim})$$

$$\frac{\Xi \vdash \phi_i\ \mathsf{tope}}{\Xi \vdash (\phi_1 \vee \phi_2)\ \mathsf{tope}}(\vee\text{-form}) \qquad \frac{\Xi \mid \Phi \vdash \phi(i)}{\Xi \mid \Phi \vdash \phi(1) \vee \phi(2)}(\vee\text{-intro})$$

$$\frac{\Xi \mid \Phi, \phi_i \vdash \chi \qquad \Xi \mid \Phi \vdash \phi_1 \vee \phi_2}{\Xi \mid \Phi \vdash \chi}(\vee\text{-elim})$$

Figure 11: Tope logic

$$\frac{\Xi \vdash t_i : I\ \mathsf{cube}}{\Xi \vdash (t_1 \equiv t_2)\ \mathsf{tope}}(\equiv\text{-form}) \qquad \frac{\Xi \vdash t : I\ \mathsf{cube}}{\Xi \vdash t \equiv t}(\equiv\text{-refl}) \qquad \frac{\Xi \mid \Phi \vdash s \equiv t}{\Xi \mid \Phi \vdash t \equiv s}(\equiv\text{-sym})$$

$$\frac{\Xi \mid \Phi \vdash r \equiv s \qquad \Xi \mid \Phi \vdash s \equiv t}{\Xi \mid \Phi \vdash r \equiv t}(\equiv\text{-trans})$$

$$\frac{\Xi, x : I \vdash \phi\ \mathsf{tope} \qquad \Xi, x : I \mid \Phi \vdash \phi[s/x] \qquad \Xi, x : I \mid \Phi \vdash s \equiv t}{\Xi, x : I \mid \Phi \vdash \phi[t/x]}(\equiv\text{-subst})$$

$$\frac{\Xi \vdash t : \mathbf{1}}{\Xi \mid \Phi \vdash t \equiv \star}(\mathbf{1}\text{-uniq}) \qquad \frac{\Xi \mid \Phi \vdash 0 \equiv 1}{\Xi \mid \Phi \vdash \bot}(\mathbb{2}\text{-uniq}) \qquad \frac{\Xi \vdash t_i : I_i}{\Xi \mid \Phi \vdash \mathsf{pr}_i \langle t_1, t_2 \rangle \equiv t_i}(\times\text{-comp})$$

$$\frac{\Xi \vdash t : I_1 \times I_2}{\Xi \mid \Phi \vdash t \equiv \langle \mathsf{pr}_1\, t, \mathsf{pr}_2\, t \rangle}(\times\text{-uniq})$$

Figure 12: Strict equality rules

For our strict equality tope the reflexivity, symmetry, and transitivity rules are introduction rules while the substitution law is an elimination rule.

$$\frac{\Xi \vdash t_i : \mathbb{2}}{\Xi \vdash (t_1 \le t_2)\ \mathsf{tope}}(\le\text{-form}) \qquad \frac{\Xi \vdash t : \mathbb{2}}{\Xi \mid \Phi \vdash t \le t}(\le\text{-refl})$$

$$\frac{\Xi \vdash t_i : \mathbb{2} \qquad \Xi \mid \Phi \vdash t_1 \le t_2 \qquad \Xi \mid \Phi \vdash t_2 \le t_3}{\Xi \mid \Phi \vdash t_1 \le t_3}(\le\text{-trans})$$

$$\frac{\Xi \vdash t_i : \mathbb{2} \qquad \Xi \mid \Phi \vdash t_1 \le t_2 \qquad \Xi \mid \Phi \vdash t_2 \le t_1}{\Xi \mid \Phi \vdash t_1 \equiv t_2}(\le\text{-antisym})$$

$$\frac{\Xi \vdash t_i : \mathbb{2}}{\Xi \mid \Phi \vdash (t_1 \le t_2) \vee (t_2 \le t_1)}(\le\text{-totl}) \qquad \frac{\Xi \vdash t : \mathbb{2}}{\Xi \mid \Phi \vdash 0 \le t}(\le\text{-min}) \qquad \frac{\Xi \vdash t : \mathbb{2}}{\Xi \mid \Phi \vdash t \le 1}(\le\text{-max})$$

Figure 13: Inequality rules

As we can see, the inequality tope has four basic constructors, two expressing 0 and 1 as the minimal and maximal terms, one expressing reflexivity, and one expressing the totality of the relation. In addition, we have one inductive introduction principle expressing transitivity. Finally, we have a new introduction rule for strict equality expressing strict anti-symmetry for the inequality tope.

Conventions for tope-notation. The operators $\le$ and $\equiv$ bind closer than $\vee$ and $\wedge$. We will take compound relations like $a \le b \equiv c$ to mean the conjunction of the individual relations, in this case $(a \le b) \wedge (b \equiv c)$, and $b \ge a$ is the same as $a \le b$. We also extend our disjunction and conjunction operations to the n-ary case using larger symbols $\bigvee$ and $\bigwedge$ respectively. As per convention, we define empty disjunctions to be trivially false $\bot$ and empty conjunctions to be trivially true $\top$.

We also have admissible rules of weakening, contraction, substitution, and cut for topes.

To produce maps from shapes to types and to ensure that the types cohere to the underlying tope logic strictly, we must also require strict type elimination rules on topes. The natural elimination rules for the "negative" topes $\top$ and $\wedge$ are already admissible by weakening and cut. However, we also require type elimination rules for the "positive" tope constructors, $\bot$ and $\vee$. In particular, these express the disjunction $\phi_1 \vee \phi_2$ as a strict pushout of $\phi_1 \wedge \phi_2 \vdash \phi_1, \phi_2$.

$$\frac{\Xi \mid \Phi \vdash \bot}{\Xi \mid \Phi \mid \Gamma \vdash \mathsf{rec}_{\bot} : A}(\bot\text{-elim}) \qquad \frac{\Xi \mid \Phi \vdash \bot \qquad \Xi \mid \Phi \vdash a : A}{\Xi \mid \Phi \mid \Gamma \vdash \mathsf{rec}_{\bot} \equiv a}(\bot\text{-comp})$$

$$\frac{\begin{array}{c}\Xi \mid \Phi \vdash \phi_1 \vee \phi_2 \\ \Xi \mid \Phi \mid \Gamma \vdash A \ \mathsf{type} \qquad \Xi \mid \Phi, \phi_i \mid \Gamma \vdash a_i : A \qquad \Xi \mid \Phi, \phi_1 \wedge \phi_2 \mid \Gamma \vdash a_1 \equiv a_2\end{array}}{\Xi \mid \Phi \mid \Gamma \vdash \mathsf{rec}_{\vee}^{\phi_1,\phi_2}(a_1, a_2) : A}(\vee\text{-elim})$$

$$\frac{\begin{array}{c}\Xi \mid \Phi \vdash \phi_1 \vee \phi_2 \\ \Xi \mid \Phi \mid \Gamma \vdash A \ \mathsf{type} \qquad \Xi \mid \Phi, \phi_i \mid \Gamma \vdash a_i : A \qquad \Xi \mid \Phi, \phi_1 \wedge \phi_2 \mid \Gamma \vdash a_1 \equiv a_2\end{array}}{\Xi \mid \Phi, \phi_i \mid \Gamma \vdash \mathsf{rec}_{\vee}^{\phi_1,\phi_2}(a_1, a_2) \equiv a_i}(\vee\text{-comp})$$

$$\frac{\Xi \mid \Phi \vdash \phi_1 \vee \phi_2 \qquad \Xi \mid \Phi \mid \Gamma \vdash a : A}{\Xi \mid \Phi \mid \Gamma \vdash \mathsf{rec}_{\vee}^{\phi_1,\phi_2}(a, a) \equiv a}(\vee\text{-uniq})$$

$$\frac{\Xi \mid \Phi \vdash s \equiv t \qquad \Xi, x : I \mid \Phi \mid \Gamma \vdash a : A}{\Xi \mid \Phi \mid \Gamma[s/x] \vdash a[s/x] \equiv a[t/x]}(\equiv\text{-}\equiv\text{-compat})^{24}$$

Figure 14: Strict type elimination for tope logic

13.3 Shapes

By a shape we mean nothing more than a tope in the singleton context of a cube. This may be formalized as

$$\frac{I \ \mathsf{cube} \qquad t : I \vdash \phi \ \mathsf{tope}}{\Xi, t : I \mid \Phi \vdash \{t : I \mid \phi\} \ \mathsf{shape}}(\mathsf{shape\text{-}form}) \qquad \frac{\Xi \vdash s : I \qquad \Xi \mid \Phi \vdash \phi[s/t]}{\Xi, t : I \mid \Phi \vdash s : \{t : I \mid \phi\}}(\mathsf{shape\text{-}intro})$$

Figure 15: Shape rules

These mainly serve as a useful abstraction to the human reader. We could equivalently assume topes in singleton contexts of cubes anywhere we assume shapes, and so we consider shapes to live on the same layer as topes.

[Φ/ϕ] Given a shape Φ we will take the lower case symbol ϕ to mean the shape's tope $\Phi \equiv \{t : I \mid \phi\}$. Note that we cannot in general go the other way as a shape is not uniquely determined by its tope. For instance, given any other cube J then both $\{t : I \mid \phi\}$ and $\{\langle t, s\rangle : I \times J \mid \phi\}$ are determined by the same tope.

[24] Note that the expression in the hypothesis is using tope-equality while the conclusion is a judgmental equality.

[$x : \Phi$] We take the notation $x : \Phi$ to be synonymous with $x : I \vdash \phi$.

[$\subseteq$] We take the relation $\Phi_0 \subseteq \Phi_1$ to be synonymous with $t : I \mid \phi_0 \vdash \phi_1$. For subshape inclusions we will prefer to start indexing at 0.

[$\varnothing$] We define the empty shapes $\varnothing$ as $\varnothing_I :\equiv \{t : I \,|\, \bot\}$ usually omitting the subscript.

[$\cap$] The intersection of two shapes in I is $\{t : I \,|\, \phi\} \cap \{t : I \,|\, \psi\} :\equiv \{t : I \,|\, \phi \wedge \psi\}$.

[$\cup$] The union of two shapes in I is $\{t : I \,|\, \phi\} \cup \{t : I \,|\, \psi\} :\equiv \{t : I \,|\, \phi \vee \psi\}$.

[$\times$] The product of two shapes where ϕ is independent of $s : J$ and ψ is independent of $t : I$ is $\{t : I \,|\, \phi\} \times \{s : J \,|\, \psi\} :\equiv \{\langle t, s\rangle : I \times J \,|\, \phi \wedge \psi\}$.

Let us give some content to the ideas by constructing a few examples. The following have their clear counterparts in simplicial sets.

Definition 13.2 (Standard shapes)**.** We define the standard n-simplex shape, its boundary and its horns for $0 \leq k \leq n$ as follows

$$\begin{aligned}
\Delta^n &:\equiv \{\langle x_1, \ldots, x_n\rangle : \mathbb{2}^n \,|\, x_n \leq \cdots \leq x_1\} \\
\partial\Delta^n &:\equiv \left\{\langle x_1, \ldots, x_n\rangle : \mathbb{2}^n \,\middle|\, \textstyle\bigvee_{0 \leq i \leq n} (0 \leq x_n \leq \cdots \leq x_{i+1} \equiv x_i \leq \cdots \leq x_1 \leq 1)\right\} \\
\Lambda^n_k &:\equiv \left\{\langle x_1, \ldots, x_n\rangle : \mathbb{2}^n \,\middle|\, \textstyle\bigvee_{0 \leq i \leq n,\, i \neq k} (0 \leq x_n \leq \cdots \leq x_{i+1} \equiv x_i \leq \cdots \leq x_1 \leq 1)\right\}
\end{aligned}$$

where the formulas at $i = 0$ and $i = n$ are the obvious extrapolations. In particular we have inclusions $\Lambda^n_k \subseteq \partial\Delta^n \subseteq \Delta^n$. To be pedantic, here are the definitions instantiated for dimensions 0, 1 and 2:

$$\Delta^0 \equiv \{\langle\rangle : \mathbf{1} \,|\, \top\} \qquad \Delta^1 \equiv \{\langle x\rangle : \mathbb{2} \,|\, \top\} \qquad \Delta^2 \equiv \left\{\langle x_1, x_2\rangle : \mathbb{2}^2 \,\middle|\, x_2 \leq x_1\right\} \qquad \partial\Delta^0 \equiv \{\langle\rangle : \mathbf{1} \,|\, \bot\}$$

$$\partial\Delta^1 \equiv \{\langle x\rangle : \mathbb{2} \,|\, x \equiv 1 \vee 0 \equiv x\} \qquad \partial\Delta^2 \equiv \left\{\langle x_1, x_2\rangle : \mathbb{2}^2 \,\middle|\, 0 \equiv x_2 \leq x_1 \vee x_2 \equiv x_1 \vee x_2 \leq x_1 \equiv 1\right\}$$

$$\Lambda^0_0 \equiv \{\langle\rangle : \mathbf{1} \,|\, \bot\} \qquad \Lambda^1_0 \equiv \{\langle x\rangle : \mathbb{2} \,|\, 0 \equiv x\} \qquad \Lambda^1_1 \equiv \{\langle x\rangle : \mathbb{2} \,|\, x \equiv 1\}$$

$$\Lambda^2_0 \equiv \left\{\langle x_1, x_2\rangle : \mathbb{2}^2 \,\middle|\, 0 \equiv x_2 \leq x_1 \vee x_2 \equiv x_1\right\} \qquad \Lambda^2_1 \equiv \left\{\langle x_1, x_2\rangle : \mathbb{2}^2 \,\middle|\, 0 \equiv x_2 \leq x_1 \vee x_2 \leq x_1 \equiv 1\right\}$$

$$\Lambda^2_2 \equiv \left\{\langle x_1, x_2\rangle : \mathbb{2}^2 \,\middle|\, x_2 \equiv x_1 \vee x_2 \leq x_1 \equiv 1\right\}.$$

13.4 Extensions

Extension types describe types of relative functions from shapes to types. An extension type is specified by an inclusion of shapes $\Phi_0 \subseteq \Phi_1$, a family of types $\Gamma, t : \Phi_1 \vdash A\ \mathsf{type}$ and a partial section $\Gamma, t : \Phi_0 \vdash a_0 : A$. The terms of the extension type are then the total sections $\Gamma, t : \Phi_1 \vdash a_1 : A$ which restrict judgmentally on Φ_0 to the partial section

$$\frac{\begin{array}{c} \{t:I\,|\,\phi_i\}\ \mathsf{shape} \qquad t:I \mid \phi_0 \vdash \phi_1 \\ \Xi \mid \Phi \vdash \Gamma\ \mathsf{ctx}^{25} \qquad \Xi, t:I \mid \Phi, \phi_1 \mid \Gamma \vdash A\ \mathsf{type} \qquad \Xi, t:I \mid \Phi, \phi_0 \mid \Gamma \vdash a : A \end{array}}{\Xi \mid \Phi \mid \Gamma \vdash \left\langle \prod_{t:I|\phi_1} A\Big|_{a}^{\phi_0} \right\rangle\ \mathsf{type}}\,(\mathsf{exten\text{-}form})$$

$$\frac{\begin{array}{c} \{t:I\,|\,\phi_i\}\ \mathsf{shape} \qquad t:I \mid \phi_0 \vdash \phi_1 \qquad \Xi \mid \Psi \vdash \Gamma\ \mathsf{ctx} \\ \Xi, t:I \mid \Phi, \phi_1 \mid \Gamma \vdash A\ \mathsf{type} \qquad \Xi, t:I \mid \Phi, \phi_i \mid \Gamma \vdash a_i : A \qquad \Xi, t:I \mid \Phi, \phi_0 \mid \Gamma \vdash a_1 \equiv a_0 \end{array}}{\Xi \mid \Psi \mid \Gamma \vdash (t:I \mid \phi_1) \mapsto a_1 : \left\langle \prod_{t:I|\phi_1} A\Big|_{a_0}^{\phi_0} \right\rangle}\,(\mathsf{exten\text{-}intro})$$

$$\frac{\begin{array}{c} \{t:I\,|\,\phi_i\}\ \mathsf{shape} \\ t:I \mid \phi_0 \vdash \phi_1 \qquad \Xi \mid \Phi \mid \Gamma \vdash f : \left\langle \prod_{t:I|\phi_1} A\Big|_{a}^{\phi_0} \right\rangle \qquad \Xi \vdash s : I \qquad \Xi \mid \Phi \vdash \phi_0[s/t] \end{array}}{\Xi \mid \Phi \mid \Gamma \vdash f(s) : A}\,(\mathsf{exten\text{-}elim})$$

$$\frac{\begin{array}{c} \{t:I\,|\,\phi_i\}\ \mathsf{shape} \\ t:I \mid \phi_0 \vdash \phi_1 \qquad \Xi \mid \Phi \mid \Gamma \vdash f : \left\langle \prod_{t:I|\phi_1} A\Big|_{a}^{\phi_0} \right\rangle \qquad \Xi \vdash s : I \qquad \Xi \mid \Phi \vdash \phi_0[s/t] \end{array}}{\Xi \mid \Phi \mid \Gamma \vdash f(s) \equiv a[s/t]}\,(\mathsf{exten\text{-}comp}_0)$$

$$\frac{\begin{array}{c} \{t:I\,|\,\phi_i\}\ \mathsf{shape} \qquad t:I \mid \phi_0 \vdash \phi_1 \qquad \Xi \mid \Psi \vdash \Gamma\ \mathsf{ctx} \qquad \Xi, t:I \mid \Phi, \phi_1 \mid \Gamma \vdash A\ \mathsf{type} \\ \Xi, t:I \mid \Phi, \phi_i \mid \Gamma \vdash a_i : A \qquad \Xi, t:I \mid \Phi, \phi_0 \mid \Gamma \vdash a_1 \equiv a_0 \qquad \Xi \vdash s : I \qquad \Xi \mid \Phi \vdash \phi_1[s/t] \end{array}}{\Xi \mid \Psi \mid \Gamma \vdash ((t:I \mid \phi_1) \mapsto a_1)(s) \equiv a_1[s/t]}\,(\mathsf{exten\text{-}comp}_1)$$

$$\frac{\{t:I\,|\,\phi_i\}\ \mathsf{shape} \qquad t:I \mid \phi_0 \vdash \phi_1 \qquad \Xi \mid \Phi \mid \Gamma \vdash f : \left\langle \prod_{t:I|\phi_1} A\Big|_{a}^{\phi_0} \right\rangle}{\Xi \mid \Phi \mid \Gamma \vdash f \equiv (t:I \mid \phi_1) \mapsto f(t)}\,(\mathsf{exten\text{-}uniq})$$

Figure 16: Extension types

a_0. Hence, informally they are strictly commuting triangles

$$\begin{tikzcd} \Phi_0 \arrow[d, hook] \arrow[dr, "a_0"] & \\ \Phi_1 \arrow[r, dashed, "a_1"'] & A. \end{tikzcd}$$

We denote this extension type by $\left\langle \prod_{t:\Phi_1} A\Big|_{a_0}^{\Phi_0} \right\rangle$ which is thus determined by the solid part of the above diagram. Their inference rules are described in Fig. 16.

In the case that A does not depend on $t:I$ in $\left\langle \prod_{t:I|\phi_1} A(t)\Big|_{a}^{\phi_0} \right\rangle$ we may use a non-dependent notation $\left\langle \{t:I\,|\,\phi_1\} \to A\Big|_{a}^{\phi_0} \right\rangle$. Furthermore, in the case that $\phi_0 \equiv \bot$ we simplify

[25]This hypothesis expresses the assumption that Γ is independent of $t:I$ and ϕ_i.

to $\prod_{t:I|\phi_1} A$ or even $\{t : I \mid \phi_1\} \to A$ in the non-dependent case. These we will simply call functions or maps from shapes to types, since they are only relative in a trivial sense. Note that the way A and a are hypothesized in our inference rules precisely make them into terms of the extension types $A : \{t : I \mid \phi_1\} \to \mathcal{U}$ (in the presence of a universe) and $a : \prod_{t:I|\phi_0} A(t)$, hence in particular our type elimination rules become basic constructors for extension terms.

14 Relating shapes and types

Having defined the type theory at hand, we now wish to develop its basics as a type theory before getting into the categorical and homotopical aspects. We separate this into three parts. First, we develop a theory of extension types and their interplay with Π- and Σ-types from Martin–Löf type theory. Then we postulate extensionality axioms for them and prove a homotopy extension property for shape inclusions. The results of these two sections are entirely due to Riehl and Shulman. Finally, we inspect the basic simplicial structure of types.

14.1 Properties of extension types

Theorem 14.1 ([RS17, Theorem 4.1])
Given a type A and a shape inclusion $\Phi_0 \subseteq \Phi_1$ with a type family $C : \Phi_1 \to A \to \mathcal{U}$ and a partial section $c : \prod_{t:\Phi_0} \prod_{a:A} C(t,a)$, then

$$\left\langle \prod_{t:\Phi_1} \left(\prod_{a:A} C(t,a) \right) \middle|_{c}^{\Phi_0} \right\rangle \simeq \prod_{a:A} \left\langle \prod_{t:\Phi_1} C(t,a) \middle|_{t \mapsto c(t,a)}^{\Phi_0} \right\rangle .$$

Moreover the inverses compose judgmentally to identities.

Do not be intimidated by the theorem statement. As is not uncommon in type theory, the type of the theorem constitutes a larger portion of the data than the proof itself. As we will see, the proof is just a simple elimination and re-introduction, with some computation to verify that the new terms restrict on the subshape to the specified partial section. We will certainly see more of this in this section.

Proof. From left to right we have the mapping $f, a, t \mapsto f(t,a)$ and from right to left we have $g, t, a \mapsto g(a,t)$. Since $f(t) \equiv c(t)$ on the subshape Φ_0, we have in particular $f(t,a) \equiv c(t,a)$. Conversely, since $g(a,t) \equiv c(t,a)$ on the subshape Φ_0, we have $t, a \mapsto g(a,t) \equiv t, a \mapsto c(t,a)$ on Φ_0 by exten-uniq. Hence our maps are well-defined. Composing left-right-left we compute

$$\begin{aligned} (g,t,a \mapsto g(a,t)) \circ (f,a,t \mapsto f(t,a)) &\equiv (f,t,a \mapsto f(t,a)) && (\mathsf{exten}\text{-comp}_1) \\ &\equiv (f \mapsto f) \equiv \mathrm{id} && (\mathsf{exten}\text{-uniq}) \end{aligned}$$

and right-left-right

$$\begin{aligned} (f,a,t \mapsto f(t,a)) \circ (g,t,a \mapsto g(a,t)) &\equiv (g,a,t \mapsto g(a,t)) && (\mathsf{exten}\text{-comp}_1) \\ &\equiv (g \mapsto g) \equiv \mathrm{id}\,. && (\mathsf{exten}\text{-uniq}) \end{aligned}$$

So the maps compose to identities judgmentally in both directions. □

Theorem 14.2 ([RS17, Theorem 4.2])
Given independent shape inclusions $\Phi_0 \subseteq \Phi_1$ and $\Psi_0 \subseteq \Psi_1$ and a type family $C : \Phi_1 \to \Psi_1 \to \mathcal{U}$

with a section c on the subshape $(\Phi_1 \times \Psi_0) \cup (\Phi_0 \times \Psi_1) \subseteq \Phi_1 \times \Psi_1$ called the ***pushout product*** *of the inclusions, then we have an equivalence*

$$\left\langle \prod_{t:\Phi_1} \left\langle \prod_{s:\Psi_1} C(t,s) \middle|^{\Psi_0}_{s\mapsto c\langle t,s\rangle} \right\rangle \middle|^{\Phi_0}_{t,s\mapsto c\langle t,s\rangle} \right\rangle \simeq \left\langle \prod_{\langle t,s\rangle:\Phi_1\times\Psi_1} C(t,s) \middle|^{(\Phi_1\times\Psi_0)\cup(\Phi_0\times\Psi_1)}_{c} \right\rangle.$$

Moreover the inverses compose judgmentally to identities.

The pushout product of shapes may be understood as the strict pushout in the following diagram

$$\begin{tikzcd} \Phi_0 \times \Psi_0 \arrow[rr, hook] \arrow[d, hook] & & \Phi_0 \times \Psi_1 \arrow[d, hook] \\ & (\Phi_1 \times \Psi_0) \cup (\Phi_0 \times \Psi_1) \arrow[ur, hook] & \\ \Phi_1 \times \Psi_0 \arrow[rr, hook] \arrow[ur, hook'] & & \Phi_1 \times \Psi_1, \end{tikzcd}$$

and we call the inclusion of the pushout product into $\Phi_1 \times \Psi_1$ the **Leibniz tensor**. Although this map is a metatheoretic thing, we will see its dual as a prominent internal construction in a later section.

Proof. We begin by showing that these types are indeed formable. That the right-hand side is well-formed is immediate, so let's consider the left-hand side. Whenever $t : \Phi_1$, we have for each $s : \Psi_0$ a term $c\langle t,s\rangle : C(t,s)$, defining a function $s \mapsto c\langle t,s\rangle : \prod_{s:\Psi_0} C(t,s)$; thus we can form $\left\langle \prod_{s:\Psi_1} C(t,s) \middle|^{\Psi_0}_{s\mapsto c\langle t,s\rangle} \right\rangle$. Now, whenever $t : \Phi_0$, we have for each $s : \Psi_1$ a term $c\langle t,s\rangle : C(t,s)$, which of course equals the first $c\langle t,s\rangle$ if ψ_0 holds, so we have the function $t,s \mapsto c\langle t,s\rangle : \prod_{t:\Phi_0} \left\langle \prod_{s:\Psi_1} C(t,s) \middle|^{\Psi_0}_{s\mapsto c\langle t,s\rangle} \right\rangle$. Thus the type on the left-hand side is formable.

The inverse equivalences are constructed similarly as in the previous theorem. From left to right we have the map $f, \langle t,s\rangle \mapsto f(t,s)$ and from right to left we have the map $g,t,s \mapsto g\langle t,s\rangle$. Composing left-right-left we compute

$$\begin{aligned} (g,t,s \mapsto g\langle t,s\rangle) \circ (f, \langle t,s\rangle \mapsto f(t,s)) &\equiv (f,t,s \mapsto f(t,s)) && (\textsf{exten-comp}_1) \\ &\equiv (f \mapsto f) \equiv \mathrm{id} && (\textsf{exten-uniq}) \end{aligned}$$

and right-left-right

$$\begin{aligned} (f, \langle t,s\rangle \mapsto f(t,s)) \circ (g,t,s \mapsto g\langle t,s\rangle) &\equiv (g, \langle t,s\rangle \mapsto g\langle t,s\rangle) && (\textsf{exten-comp}_1) \\ &\equiv (g \mapsto g) \equiv \mathrm{id}. && (\textsf{exten-uniq}) \end{aligned}$$

Hence the maps compose to identities judgmentally in both directions. □

Corollary 14.3. *If X is either a shape or a type and $\Phi_0 \subseteq \Phi_1$ is a shape inclusion with a type family $C : \Phi_1 \to X \to \mathcal{U}$ with partial section $c : \prod_{t:\Phi_0} \prod_{x:X} C(t,x)$ then we have the equivalence*

$$\left\langle \prod_{t:\Phi_1} \left(\prod_{x:X} C(t,x) \right) \middle|^{\Phi_0}_{c} \right\rangle \simeq \prod_{x:X} \left\langle \prod_{t:\Phi_1} C(t,x) \middle|^{\Phi_0}_{c} \right\rangle.$$

Proof. If X is a type then this is the exact statement of Theorem 14.1, and if X is a shape we may apply Theorem 14.2 twice to the inclusion $\varnothing \subseteq X$ to get the equivalences

$$\left\langle \prod_{t:\Phi_1}\left(\prod_{x:X} C(t,x)\right)\middle|^{\Phi_0}_{c} \right\rangle \simeq \left\langle \prod_{\langle t,s\rangle:\Phi_1\times X} C(t,s)\middle|^{\Phi_0\times X}_{\langle t,s\rangle\mapsto c(t,s)} \right\rangle \simeq \prod_{x:X}\left\langle \prod_{t:\Phi_1} C(t,x)\middle|^{\Phi_0}_{t\mapsto c(t,x)} \right\rangle. \qquad \square$$

Theorem 14.4 (Type-theoretic axiom of relative choice, [RS17, Theorem 4.3])
Given a shape inclusion $\Phi_0 \subseteq \Phi_1$ and a type family $C : \Phi_1 \to \mathcal{U}$ with partial section $c : \prod_{t:\Phi_0} C(t)$ and further a family $D : \prod_{t:\Phi_1}(C(t) \to \mathcal{U})$ with partial section $d : \prod_{t:\Phi_0} D(t, c(t))$, then we have the following equivalence

$$\left\langle \prod_{t:\Phi_1}\left(\sum\nolimits_{x:C(t)} D(t,x)\right)\middle|^{\Phi_0}_{(c,d)} \right\rangle \simeq \sum\nolimits_{f:\left\langle \prod_{t:\Phi_1} C(t)\middle|^{\Phi_0}_{c}\right\rangle} \left\langle \prod_{t:\Phi_1} D(t, f(t))\middle|^{\Phi_0}_{d} \right\rangle.$$

Moreover, the inverses compose judgmentally to identities.

This theorem is a straightforward adaptation of the type-theoretic theorem of choice as seen in Theorem 3.3 to extension types. In the case that $\Phi_0 \equiv \varnothing$ we recover the reading: "The existence of terms $x : C(t)$ such that $D(t,x)$ over all of Φ_1 is equivalent to the existence of a choice function $f : \prod_{t:\Phi_1} C(t)$ such that $D(t, f(t))$ over all of Φ_1."

$$\prod_{t:\Phi_1}\left(\sum_{x:C(t)} D(t,x)\right) \simeq \sum_{f:\prod_{t:\Phi_1} C(t)}\left(\prod_{t:\Phi_1} D(t, f(t))\right)$$

Now in the general case, we additionally have that we may fix partial sections c and d such that the choice function restricts to the partial sections judgmentally.

Proof. From left to right we have the map $h \mapsto (t \mapsto \mathsf{pr}_1\, h(t), t \mapsto \mathsf{pr}_2\, h(t))$ and from right to left we have the map $(f,g), t \mapsto (f(t), g(t))$. Their compositions yield

$$\begin{aligned}
&((f,g), t \mapsto (f(t), g(t))) \circ (h \mapsto (t \mapsto \mathsf{pr}_1\, h(t), t \mapsto \mathsf{pr}_2\, h(t))) \\
&\qquad \equiv (h, s \mapsto ((t \mapsto \mathsf{pr}_1\, h(t))(s), (t \mapsto \mathsf{pr}_2\, h(t))(s))) && (\Pi\text{-comp}) \\
&\qquad \equiv (h, s \mapsto (\mathsf{pr}_1\, h(s), \mathsf{pr}_2\, h(s))) && (\mathsf{exten}\text{-comp}_1) \\
&\qquad \equiv (h, s \mapsto h(s)) && (\Sigma\text{-uniq}) \\
&\qquad \equiv (h \mapsto h) \equiv \mathrm{id} && (\Pi\text{-uniq})
\end{aligned}$$

and

$$\begin{aligned}
&(h \mapsto (t \mapsto \mathsf{pr}_1\, h(t), t \mapsto \mathsf{pr}_2\, h(t))) \circ ((f,g), t \mapsto (f(t), g(t))) \\
&\equiv ((f,g) \mapsto (s \mapsto \mathsf{pr}_1(t \mapsto (f(t), g(t)))(s), s \mapsto \mathsf{pr}_2(t \mapsto (f(t), g(t)))(s))) && (\Pi\text{-comp}) \\
&\equiv ((f,g) \mapsto (s \mapsto f(s), s \mapsto g(s))) && (\Sigma\text{-comp}) \\
&\equiv ((f,g) \mapsto (f,g)) \equiv \mathrm{id}. && (\mathsf{exten}\text{-uniq})
\end{aligned}$$

$\square$

We state two more properties without proof.

Theorem 14.5 ([RS17, Theorem 4.4])
Given the double shape inclusion $\Phi_0 \subseteq \Phi_1 \subseteq \Phi_2$ with a type family $C : \Phi_2 \to \mathcal{U}$ and partial section $c : \prod_{t:\Phi_0} C(t)$, then

$$\left\langle \prod_{t:\Phi_2} C(t) \middle|^{\Phi_0}_{c} \right\rangle \simeq \sum\nolimits_{f:\left\langle \prod_{t:\Phi_1} C(t)\middle|^{\Phi_0}_{c}\right\rangle} \left\langle \prod_{t:\Phi_2} C(t) \middle|^{\Phi_1}_{f} \right\rangle.$$

Theorem 14.6 ([RS17, Theorem 4.5])
Given two shapes in the same cube $t : I \vdash \phi$ tope and $t : I \vdash \psi$ tope and a type family over their union $C : \{t : I \mid \phi \vee \psi\} \to \mathcal{U}$ with a section over ψ, $c : \prod_{t:I|\psi} C(t)$, then

$$\left\langle \prod_{t:I|\phi\vee\psi} C(t) \middle|^{\psi}_{c} \right\rangle \simeq \left\langle \prod_{t:I|\phi} C(t) \middle|^{\phi\wedge\psi}_{c} \right\rangle.$$

14.2 Extensionality of extensions

We also require an extensionality principle for extension types. This principle plays a particularly important role as it implies the homotopy extension property for extensions. There are multiple ways to formulate an extensionality principle for relative functions inspired by equivalent principles for functions of types. However, it is not currently known if they are equivalent for relative functions. We look at a weak formulation and then the formulation we assume for the rest of the text.

Assume given an extension term $f : \left\langle \prod_{t:\Phi_1} A(t) \middle|^{\Phi_0}_{a} \right\rangle$, then we have the diagonal term

$$t \mapsto \mathsf{refl}_{f(t)} : \left\langle \prod_{t:\Phi_1} f(t) = f(t) \middle|^{\Phi_0}_{t\mapsto \mathsf{refl}_{a(t)}} \right\rangle,$$

so by path induction we obtain the following map from *identity to relative homotopy*

$$\mathsf{id\text{-}to\text{-}rel\text{-}htp} \ : \prod_{f,g:\left\langle \prod_{t:\Phi_1} A(t)\middle|^{\Phi_0}_{a}\right\rangle} (f = g) \to \left\langle \prod_{t:\Phi_1} f(t) = g(t) \middle|^{\Phi_0}_{t\mapsto \mathsf{refl}_{a(t)}} \right\rangle \quad \mathsf{id\text{-}to\text{-}rel\text{-}htp}(\mathsf{refl}_f) :\equiv t \mapsto \mathsf{refl}_{f(t)}.$$

Axiom 6 (Weak relative function extensionality)**.** *The map* $\mathsf{id\text{-}to\text{-}rel\text{-}htp}$ *is an equivalence.*

Axiom 7 (Relative function extensionality)**.** *Given a type family $A : \Phi_1 \to \mathcal{U}$ such that each fiber $A(t)$ is contractible, then every extension type $\left\langle \prod_{t:\Phi_1} A(t) \middle|^{\Phi_0}_{a} \right\rangle$ is contractible.*

Theorem 14.7 ([RS17, Proposition 4.8(i)])
Relative function extensionality implies weak relative function extensionality.

Proof. To prove $\mathsf{id\text{-}to\text{-}rel\text{-}htp}$ is an equivalence it suffices to show that the induced map on total spaces

$$\left(\sum_{g:\left\langle \prod_{t:\Phi_1} A(t)\middle|^{\Phi_0}_{a}\right\rangle} f = g \right) \to \sum_{g:\left\langle \prod_{t:\Phi_1} A(t)\middle|^{\Phi_0}_{a}\right\rangle} \left\langle \prod_{t:\Phi_1} f(t) = g(t) \middle|^{\Phi_0}_{t\mapsto \mathsf{refl}_{a(t)}} \right\rangle$$

is an equivalence. But the left-hand side is a based path space and hence contractible, so it remains to show that the codomain is contractible. By Theorem 14.4 it is equivalent to $\left\langle \prod_{t:\Phi_1} \sum_{y:A(t)} (f(t) = y) \middle|^{\Phi_0}_{t \mapsto (a(t), \mathsf{refl}_{a(t)})} \right\rangle$. But each $\sum_{y:A(t)} (f(t) = y)$ is contractible since it is a based path space, so by relative function extensionality we have our result. $\square$

From now on we assume the strong formulation of relative function extensionality. From it, we may in particular establish the homotopy extension property.

Proposition 14.8 (Homotopy extension property (HEP), [RS17, Proposition 4.10])**.** *Assuming relative function extensionality, given a shape inclusion $\Phi_0 \subseteq \Phi_1$ and a type family $A : \Phi_1 \to \mathcal{U}$ with a total section $b : \prod_{t:\Phi_1} A(t)$ and a partial section $a : \prod_{t:\Phi_0} A(t)$ and a homotopy on the subdomain $H : \prod_{t:\Phi_0} a(t) = b(t)$, then we have extensions of a and H to the whole of Φ_1: $a' : \left\langle \prod_{t:\Phi_1} A(t) \middle|^{\Phi_0}_{a} \right\rangle$ and $H' : \left\langle \prod_{t:\Phi_1} a'(t) = b(t) \middle|^{\Phi_0}_{t \mapsto H(t)} \right\rangle$.*

The homotopy extension property may be encoded informally fiberwise in the following left-hand diagram. The strict commutativity properties are encoded in the two right-hand diagrams. Note that the maps a and a' are already determined up to homotopy from the other maps.

$$\begin{tikzcd}
\Phi_0 \arrow[r, "H"] \arrow[d, hook] \arrow[dr, "a"' near end] & A^{\mathrm{I}} \arrow[d, "\mathrm{ev}_0"] \\
\Phi_1 \arrow[ur, dashed, "H'" near start] \arrow[r, dashed, shift left, "a'"] \arrow[r, shift right, "b"'] & A
\end{tikzcd}
\qquad
\begin{tikzcd}
\Phi_0 \arrow[r, "H"] \arrow[d, hook] \arrow[dr, "a"' near end] & A^{\mathrm{I}} \arrow[d, "\mathrm{ev}_0"] \\
\Phi_1 \arrow[ur, dashed, "H'" near start] \arrow[r, dashed, "a'"] & A
\end{tikzcd}
\quad
\begin{tikzcd}
\Phi_0 \arrow[r, "H"] \arrow[d, hook] & A^{\mathrm{I}} \arrow[d, "\mathrm{ev}_1"] \\
\Phi_1 \arrow[ur, dashed, "H'" near start] \arrow[r, "b"'] & A
\end{tikzcd}$$

Proof. The extension type $\left\langle \prod_{t:\Phi_1} \sum_{y:A(t)} y = b(t) \middle|^{\Phi_0}_{t \mapsto (a(t), e(t))} \right\rangle$ is contractible by relative function extensionality, and so in particular it is inhabited. Applying the type-theoretic axiom of choice we obtain a' and H'. $\square$

14.3 Arrows

Since types are simplicial ∞-groupoids, they already come equipped with quite a bit of structure out of the box. In this section, we take a look at their simplicial structure without adding assumptions on the types under consideration. In particular, we define the n-cells and see that they are preserved by functions by definition, as with types and identity terms in homotopy type theory. We may also define higher-order arrows, i.e. natural transformations, and define a horizontal composition operation on them. Hence, without introducing the internal notion of categories in the type theory, the Segal or Rezk types, there are already many categorical notions that are well-defined and have meaningful interpretations. We start our considerations with non-dependent simplices but will return to the dependent case later.

Definition 14.9 (Hom-types)**.** Given two terms $x, y : A$, the type of **arrows** or **directed edges** or (**homo-**)**morphisms** or **1-cells** from x to y in A is

$$\mathsf{hom}_A(x, y) :\equiv \left\langle \Delta^1 \to A \middle|^{\partial \Delta^1}_{[x,y]} \right\rangle \equiv \langle x \dashrightarrow y \rangle_A$$

where $[x,y] :\equiv \mathsf{rec}_{\vee}^{t\equiv 0, t\equiv 1}(x,y)$. As an alternative notation, we will also use $x \to_A y$ for the hom-type $\mathsf{hom}_A(x,y)$. Observe that a hom-term f acts like a function $\Delta^1 \to A$ with strict endpoints $f(0) \equiv x$ and $f(1) \equiv y$ (due to $\mathsf{exten\text{-}comp}_1$).

This notion generalizes to the n-dimensional case as follows.

Definition 14.10 (n-cells)**.** Given a boundary diagram $\delta : \partial\Delta^n \to A$ we define the type of n**-cells** with boundary δ to be

$$\mathsf{hom}_A^n\,\delta :\equiv \left\langle \Delta^n \to A\middle|_{\delta}^{\partial\Delta^n} \right\rangle .$$

These are the n-dimensional simplices in the categorical direction. For categories they may be interpreted as n-dimensional coherences and can diagramatically be depicted as an n-dimensional "filler." For instance we will admit associativity of composition in Segal types as a 3-dimensional coherence, and will see that identity in Segal types coincide with a degenerate family of 2-cells.

Due to the frequent occurence of 2-cells, we also introduce a notation for their boundaries. Given a triangle of arrows $f : x \to_A y$, $g : y \to_A z$ and $h : x \to_A z$, then

$$[f,g;h] :\equiv \mathsf{rec}_{\vee}^{0\equiv s\leq t, s\leq t\equiv 1, s\equiv t}(f,g,h) : \partial\Delta^2 \to A$$

taking the ternary recursor to mean either choice of iterated recursion. We note that f and g are written in diagramatic order. For instance, a term of $\mathsf{hom}_A^2[f,g;h]$, usually denoted $\mathsf{hom}_A^2(f,g;h)$, witnesses h as a composite of g after f.

Definition 14.11 (Identity arrows)**.** For every type we have a map from paths to arrows defined by path induction given on reflexivities by the constant arrows

$$\mathsf{id\text{-}to\text{-}arr}_A : \prod_{x,y:A} (x =_A y) \to (x \to_A y) \qquad \mathsf{id\text{-}to\text{-}arr}_A(\mathsf{refl}_x) :\equiv (t \mapsto x) \equiv: \mathsf{id}_x\,.$$

These are called the **identity arrows**.

Given any arrow f in a type, we may form the following degenerate 2-cells,

$$\begin{tikzcd} & x \arrow[dr, "f"] & \\ x \arrow[ur, equal] \arrow[rr, "f"'] & & y \end{tikzcd} \qquad \text{and} \qquad \begin{tikzcd} & y \arrow[dr, equal] & \\ x \arrow[ur, "f"] \arrow[rr, "f"'] & & y. \end{tikzcd}$$

witnessing the unit laws when a composition operation is defined. These are constructed from f using the following lambda abstractions

$$\langle s,t\rangle \mapsto f(t) \quad \text{and} \quad \langle s,t\rangle \mapsto f(s).$$

Remark 14.12 (Functions preserve simplicial structure)**.** Let us demonstrate that any function defined in this type theory preserves the simplicial structure of the types.

Assume given a shape inclusion $\Phi_0 \subseteq \Phi_1$, type families $A, B : \Phi_1 \to \mathcal{U}$ and a partial section $a : \prod_{t:\Phi_0} A(t)$. Then given any fibered mapping $f : \prod_{t:\Phi_1}(A(t) \to B(t))$ we may apply exten-elim and exten-intro to construct a mapping of extension types given by postcomposing with f:

$$f \circ - :\equiv g, t \mapsto f(g(t)) : \left\langle \prod_{t:\Phi_1} A(t) \middle|_{a}^{\Phi_0} \right\rangle \to \left\langle \prod_{t:\Phi_1} B(t) \middle|_{t \mapsto f(a(t))}^{\Phi_0} \right\rangle .$$

This demonstrates the important fact that functions preserve simplicial structure. In particular, $f \circ -$ maps n-cells in A to n-cells in B. It also sends constant arrows to constant arrows definitionally which we can see by the following computation:

$$f \circ \mathsf{id}_x \equiv t \mapsto f((s \mapsto x)(t)) \equiv t \mapsto f(x) \equiv \mathsf{id}_{f(x)} .$$

Since function types and extension types are *types* on the same level as other types, they are also simplicial ∞-groupoids. This reflects the fact that simplicial ∞-groupoids *exponentiate* or "admit internal homs." When taking this viewpoint (or in case we are at a shortage for space) we may denote the function/extension type $X \to A$ as A^X.

Arrows in function types form a particularly important example. We call these arrows **natural transformations**.

Definition 14.13 (Natural transformation)**.** Given functions $f, g : X \to B$ where X is either a shape or a type, we refer to terms of the type $f \Rightarrow g :\equiv \mathsf{hom}_{X \to B}(f, g)$ as **natural transformations from** f **to** g. Given such a natural transformation we say its domain is X, its codomain is B, its source is f and its target is g.

From a natural transformation α we may recover its **component at** $x : X$ as

$$\alpha_x :\equiv t \mapsto \alpha(t, x) : \mathsf{hom}_B(f(x), g(x))$$

and moreover the total component mapping $\alpha_{(-)}$ is an equivalence.

Proposition 14.14 (Natural transformation extensionality, [RS17, Proposition 6.3])**.** *Natural transformations are determined by their components. Given functions $f, g : X \to B$ where X is either a shape or a type, then*

$$\mathsf{hom}_{X \to B}(f, g) \simeq \prod_{x:X} \mathsf{hom}_B(f(x), g(x)).$$

Proof. By expanding definitions we obtain the following type signature of the component mapping

$$\left\langle \Delta^1 \to (X \to B) \middle|_{[f,g]}^{\partial\Delta^1} \right\rangle \to \prod_{x:X} \left\langle \Delta^1 \to B \middle|_{[f(x),g(x)]}^{\partial\Delta^1} \right\rangle$$

At which point we may apply Corollary 14.3 to get our result. □

Hence we may decide to define natural transformations *componentwise*, in which case we implicitly apply the inverse of this equivalence.

We do not recover 2-cells as a special case of natural transformations as when the arrows $f, g : x \to_B y$ are considered as terms of $\Delta^1 \to B$ the end-points are free; natural transformations do not, in general, stay fixed at them. There is however a way of constructing natural transformations from 2-cells, as we will see in the proof of Theorem 15.8.

Construction 14.15 (Horizontal composites)**.** Given a diagram of types where X may be a shape

$$\begin{tikzcd} X \arrow[r, bend left, "f"] \arrow[r, bend right, "f'"'] \arrow[r, phantom, "\Downarrow\alpha"] & A \arrow[r, bend left, "g"] \arrow[r, bend right, "g'"'] \arrow[r, phantom, "\Downarrow\beta"] & B \end{tikzcd}$$

Then there is a natural transformation $\beta \circ_{\mathrm{h}} \alpha : g \circ f \Rightarrow g' \circ f'$ called **the horizontal composite of** α **and** β. We define it componentwise as

$$(\beta \circ_{\mathrm{h}} \alpha)_x :\equiv \beta_{\alpha_x} \equiv (t \mapsto \beta(t, \alpha(t,x))) : (g \circ f)(x) \to_B (g' \circ f')(x).$$

We abusively use juxtaposition $\beta\alpha$ for horizontal composition, and reserve $\beta \circ \alpha$ for the to-be-defined vertical composite. As special cases of horizontal composition we recover the whiskering operations $\beta f :\equiv \beta \circ_{\mathrm{h}} \mathsf{id}_f$ and $g\alpha :\equiv \mathsf{id}_g \circ_{\mathrm{h}} \alpha$.

The horizontal composite is identified with the diagonal edge in the *Gray interchanger* diagram

$$\begin{tikzcd} g \circ f \arrow[r, "g\alpha"] \arrow[d, "\beta f"'] \arrow[dr, "\beta\alpha"] & g \circ f' \arrow[d, "\beta f'"] \\ g' \circ f \arrow[r, "g'\alpha"'] & g' \circ f' \end{tikzcd} \qquad \text{abstracted as} \quad \langle t, s \rangle, x \mapsto \beta(s, \alpha(t,x))$$

witnessing $\beta\alpha$ as both a composite of $g\alpha$ with $\beta f'$ and a composite of βf with $g'\alpha$.

15 Category theory in simplicial homotopy type theory

15.1 Segal types

We now move on to introduce the internal precategories of our theory (in terminology consistent with [UF13]). This is in the sense that the system of identification of terms of the type may be weaker than isomorphism, and so these types may also be thought of as flagged categories. We call them Segal types since they semantically correspond to the Segal spaces in the motivating model of bisimplicial sets. The Segal types are internally characterized by having homotopy unique composites. Surprisingly, from this, all other coherence laws follow.

We begin by defining yet another type of simplices:

Definition 15.1 (Composition of arrows)**.** The type of **compositions of arrows** $x \xrightarrow{f} y \xrightarrow{g} z$ is the extension type

$$\mathsf{comp}_A(f,g) :\equiv \left\langle \Delta^2 \to A \middle|^{\Lambda_1^2}_{[f,g]} \right\rangle \equiv \left\langle \begin{tikzcd} & y \arrow[dr, "g"] & \\ x \arrow[ur, "f"] \arrow[rr, dashed] & & z \end{tikzcd} \right\rangle_A.$$

From a composition term we may recover an arrow $x \to z$ by restricting along the meta-theoretic embedding $t \mapsto \langle t, t \rangle : \Delta^1 \hookrightarrow \Delta^2$. We call this arrow *a composite of f and*

g. Given such an arrow h we may additionally recover a *witness of composition*, meaning a term of $\mathsf{hom}^2_A(f,g;h)$. Given a composition $\phi : \mathsf{comp}_A(f,g)$, using application and reabstraction we may consider it as a term of the extension type $\left\langle \Delta^2 \to A \middle|^{\partial\Delta^2}_{[f,g,t\mapsto\phi\langle t,t\rangle]} \right\rangle$. Remember, since an arbitrary type in our theory is just a simplicial ∞-groupoid we may have many non-homotopic candidates for a composite.

Remark 15.2. In Riehl and Shulman's original paper [RS17], they define the type of compositions as the dependent sum

$$\sum_{h:\mathsf{hom}_A(x,z)} \mathsf{hom}^2_A\left(\begin{tikzcd}[sep=small] & y \arrow[dr, "g"] & \\ x \arrow[ur, "f"] \arrow[rr, "h"'] & & z \end{tikzcd}\right)$$

and prove it to be equivalent to the type we use as definition. It is not hard to see that the maps constructing the composite and witness of composition give rise to an equivalence of types. However, we have a shorter route without the need to compute inverses:

$$\begin{aligned} \left\langle \Delta^2 \to A \middle|^{\Lambda^2_1}_{[f,g]} \right\rangle &\simeq \textstyle\sum_{\delta:\left\langle \partial\Delta^2 \to A \middle|^{\Lambda^2_1}_{[f,g]} \right\rangle} \left\langle \Delta^2 \to A \middle|^{\partial\Delta^2}_{\delta} \right\rangle && \text{(Theorem 14.5)} \\ &\simeq \textstyle\sum_{h:\left\langle \Delta^1 \to A \middle|^{\partial\Delta^1}_{[x,z]} \right\rangle} \left\langle \Delta^2 \to A \middle|^{\partial\Delta^2}_{[f,g,h]} \right\rangle && \text{(Theorem 14.4)} \\ &\equiv \sum_{h:\mathsf{hom}_A(x,z)} \mathsf{hom}^2_A\left(\begin{tikzcd}[sep=small] & y \arrow[dr, "g"] & \\ x \arrow[ur, "f"] \arrow[rr, "h"'] & & z \end{tikzcd}\right). \end{aligned}$$

The choice of definition is mostly a matter of taste, and we stick with the extension type formulation.

Definition 15.3 (Segal type)**.** A **Segal type** is a type B such that the type of compositions $\mathsf{comp}_B(f,g)$ is contractible for every pair of composable arrows f, g in B. We denote their composite arrow by $g \circ f$ and their witness of composition by $\mathsf{comp}_{f,g}$ which thus is unique up to homotopy.

$$\mathsf{isSegal}\, B :\equiv \prod_{x,y,z:B} \prod_{\substack{f:x\to_B y,\\ g:y\to_B z}} \mathsf{isContr}\,(\mathsf{comp}_B(f,g))$$

At first glance, it may seem like our Segal condition semantically corresponds to the Segal map

$$B_2 \to B_1 \times_{B_0} B_1$$

being an equivalence of simplicial sets. However, since our condition is defined internally to the theory it instead corresponds to the bisimplicial map

$$B^{\Delta^2} \to B^{\Delta^1} \times_{B^{\Delta^0}} B^{\Delta^1} \simeq B^{\Lambda^2_1}$$

being an equivalence of bisimplicial sets. This condition was conjectured by Joyal to be equivalent to the usual Segal condition, and this was proven true by Riehl and Shulman [RS17, Proposition A.21].

Theorem 15.4 ([RS17, Theorem 5.5])
A type B is Segal iff the restriction map $(\Delta^2 \to B) \to (\Lambda_1^2 \to B)$ is an equivalence.

Proof. the type B is Segal iff each $\left\langle \Delta^2 \to A \middle|^{\Lambda_1^2}_{\delta} \right\rangle$ is contractible. Moreover, we have the equivalence

$$(\Delta^2 \to A) \simeq \sum_{\delta : \Lambda_1^2 \to A} \left\langle \Delta^2 \to A \middle|^{\Lambda_1^2}_{\delta} \right\rangle \qquad \text{(Theorem 14.5)}$$

Now, since the projection from a total space is an equivalence exactly when all the fibers are contractible, the result follows. □

Corollary 15.5 ([RS17, Corollary 5.6])**.** *Given a type or shape X and a type family $B : X \to \mathcal{U}$ such that each $B(x)$ is Segal, then the dependent function type $\prod_{x:X} B(x)$ is Segal. In particular, when B is independent of X, the type B^X is Segal.*

Proof. We may apply Corollary 14.3 to rearrange arguments and get

$$(\Delta^2 \to \prod_{x:X} B(x)) \simeq \prod_{x:X} (\Delta^2 \to B(x)) \quad \text{and} \quad (\Lambda_1^2 \to \prod_{x:X} B(x)) \simeq \prod_{x:X} (\Lambda_1^2 \to B(x)).$$

Since dependent products preserve fiberwise equivalences (using function extensionalty) the result follows from Theorem 15.4. □

Remark 15.6 (Vertical composition)**.** For a Segal type B and any shape or type X, composition on B^X is a vertical composition operation on natural transformations

$$\begin{tikzcd} X \arrow[r, bend left=60, "f"{name=F}] \arrow[r, "g"{description, name=G}] \arrow[r, bend right=60, "h"'{name=H}] & B. \arrow[from=F, to=G, phantom, "\Downarrow\alpha"] \arrow[from=G, to=H, phantom, "\Downarrow\beta"] \end{tikzcd}$$

In particular, the components of $\beta \circ \alpha$ may be identified with the composite of their respective components

$$(\beta \circ \alpha)_x = \beta_x \circ \alpha_x.$$

Proposition 15.7 ([RS17, Proposition 5.8])**.** *In a Segal type B, identity arrows are units: given any $f : x \to_B y$, then $f \circ \mathsf{id}_x = f$ and $\mathsf{id}_y \circ f = f$.*

Proof. For the left unit law we have the canonical witness

$$\langle t, s \rangle \mapsto f(s) : \mathsf{comp}_B(\mathsf{id}_x, f).$$

To see that it has the correct boundary, observe that

$$(\langle s, r \rangle \mapsto f(r)) \circ (t \mapsto \langle t, 0 \rangle) \equiv t \mapsto ((\langle s, r \rangle \mapsto f(r))\langle t, 0 \rangle) \equiv t \mapsto f(0) \equiv t \mapsto x \equiv \mathsf{id}_x$$

and

$$(\langle s, r \rangle \mapsto f(r)) \circ (t \mapsto \langle 1, t \rangle) \equiv t \mapsto ((\langle s, r \rangle \mapsto f(r))\langle 1, t \rangle) \equiv t \mapsto f(t) \equiv f$$

and finally, their composite arrow is

$$(\langle s,r\rangle \mapsto f(r)) \circ (t \mapsto \langle t,t\rangle) \equiv t \mapsto ((\langle s,r\rangle \mapsto f(r))\langle t,t\rangle) \equiv t \mapsto f(t) \equiv f.$$

By a similar computation we conclude that $\langle t,s\rangle \mapsto f(s)$ witnesses the right unit law. $\square$

Theorem 15.8 (Associativity for Segal types, [RS17, Proposition 5.9])
Given a diagram $x \xrightarrow{f} y \xrightarrow{g} z \xrightarrow{h} w$ *in a Segal type* B*, then* $(h \circ g) \circ f = h \circ (g \circ f)$.

Proof. Assume given cube variables $t, s : \mathbb{2}$. By totality of the simplicial relation, we have the entailments $\top \vdash (t \le s) \vee (s \le t) \vdash \top$. Hence we may observe that the shapes $\Delta^1 \times \Delta^1$ and $\Delta^2 \cup_{\Delta^1_1} \Delta^2$ are equivalent.

Using this, we may consider the following square as a natural transformation

$$\begin{tikzcd}
x \arrow[r, "f"] \arrow[d, "f"'] \arrow[dr, "g\circ f" description] \arrow[dr, phantom, shift left=3, "\mathsf{comp}_{f,g}"] \arrow[dr, phantom, shift right=3, "\mathsf{comp}_{f,g}"'] & y \arrow[d, "g"] \\
y \arrow[r, "g"'] & z
\end{tikzcd}$$

i.e. as an arrow $\mathsf{comp}_{f,g} : f \to_{B^{\Delta^1}} g$. Similarly we have an arrow $\mathsf{comp}_{g,h} : g \to_{B^{\Delta^1}} h$.

By Corollary 15.5 we have that B^{Δ^1} is Segal, and so by Remark 15.2 the type

$$\sum_{p:\mathsf{hom}_{B^{\Delta^1}}(f,h)} \mathsf{hom}^2_{B^{\Delta^1}}(\mathsf{comp}_{f,g}, \mathsf{comp}_{g,h}; p)$$

is contractible. In particular this means it is inhabited by some pair (p,q) where q is some function $\Delta^2 \times \Delta^1 \to B$. There is a function that picks out the "middle shuffle"

$$\langle t_1, t_2, t_3\rangle \mapsto \langle\langle t_1, t_3\rangle, t_2\rangle : \Delta^3 \to \Delta^2 \times \Delta^1.$$

This gives us a 3-simplex in B,

$$\begin{tikzcd}
 & z \arrow[rr, "h"] & & w \\
x \arrow[ur, "g\circ f"] \arrow[urrr] \arrow[rr, "f"'] & & y \arrow[ul, "g"'] \arrow[ur, "h\circ g"'] &
\end{tikzcd}$$

The top and bottom faces are parametrized by the restrictions

$$\langle t_1, t_2\rangle \mapsto \langle t_1, t_1, t_2\rangle : \Delta^2 \to \Delta^3 \quad \text{and} \quad \langle t_1, t_2\rangle \mapsto \langle t_1, t_2, t_2\rangle : \Delta^2 \to \Delta^3$$

with common edge, the diagonal,

$$t \mapsto \langle t,t,t\rangle : \Delta^1 \to \Delta^3.$$

This edge defines an arrow $x \to_B w$ for which the restrictions along the two faces yield witnesses that it is the composite $h \circ (g \circ f)$ and the composite $(h \circ g) \circ f$ respectively. $\square$

Digression 15.9. We could in even more general terms speak about types equipped with some binary operation $- \bullet - : \prod_{(x,y,z:A)} \prod_{(f:x \to_A y)} \prod_{(g:y \to_A z)} \mathsf{comp}_A(f,g)$. This would correspond to generalizing from having an equivalence induced by $\Lambda^2_1 \subseteq \Delta^2$ to only admitting a section. For such an operation we still maintain the unit laws, but associativity is no longer clear. This would allow us to speak of types with multiple composition operations on their arrows.

Proposition 15.10 (Functions are functors, [RS17, Proposition 6.1])**.** *Given a function between Segal types $F : A \to B$ and any two composable arrows f and g in A, then*

$$F(g \circ f) = Fg \circ Ff.$$

Proof. Since functions are simplicial morphisms we have $F(g \circ f) : \mathsf{comp}_B(Ff, Fg)$, and by contractibilty of this type we have $F(g \circ f) = Fg \circ Ff$. □

Homotopy. In simplicial types we have two competing notions of sameness for arrows. We have paths between arrows $f =_{(x \to_A y)} g$ and we have 2-simplices between them

$$\begin{tikzcd} & x \arrow[dr, "f"] & \\ x \arrow[ur, equal] \arrow[rr, "g"'] & & y \end{tikzcd} \quad \text{or} \quad \begin{tikzcd} & y \arrow[dr, equal] & \\ x \arrow[ur, "f"] \arrow[rr, "g"'] & & y. \end{tikzcd}$$

Luckily for us, these notions are equivalent in Segal types.

Proposition 15.11 (Homotopy is homotopy, [RS17, Proposition 5.10])**.** *Let $f, g : x \to_B y$ in a Segal type B. Then we have the following equivalences via natural maps*

$$\mathsf{hom}^2_B\left(\begin{tikzcd} & x \arrow[dr, "f"] & \\ x \arrow[ur, equal] \arrow[rr, "g"'] & & y \end{tikzcd}\right) \simeq (f =_{(x \to_B y)} g) \simeq \mathsf{hom}^2_B\left(\begin{tikzcd} & y \arrow[dr, equal] & \\ x \arrow[ur, "f"] \arrow[rr, "g"'] & & y \end{tikzcd}\right).$$

Proof. By duality we only consider the right-side case. We may define the following map by path induction for any type B

$$\mathsf{id\text{-}to\text{-}2\text{-}cell}_{f,g} : (f =_{(x \to_B y)} g) \to \mathsf{hom}^2_B\left(\begin{tikzcd} & y \arrow[dr, equal] & \\ x \arrow[ur, "f"] \arrow[rr, "g"'] & & y \end{tikzcd}\right), \quad \mathsf{id\text{-}to\text{-}2\text{-}cell}_{f,f}(\mathsf{refl}_f) :\equiv (\langle t, s \rangle \mapsto f(t)).$$

To prove it is a fiberwise equivalence is equivalent [UF13, Theorem 4.7.7] to proving the induced map on total spaces is an equivalence

$$\left(\sum_{g:x \to_B y} f = g\right) \to \sum_{g:x \to_B y} \mathsf{hom}^2_B\left(\begin{tikzcd} & y \arrow[dr, equal] & \\ x \arrow[ur, "f"] \arrow[rr, "g"'] & & y \end{tikzcd}\right).$$

The left-hand side is contractible by definition, and the right-hand side is equivalent to the type of compositions $\mathsf{comp}_B(f, \mathsf{id}_y)$ by Remark 15.2, hence it is contractible by the Segal condition. □

This property easily extends to general triangles.

Proposition 15.12. *Given a triangle of arrows f, g, h in a Segal type B, then*

$$(g \circ f = h) \simeq \mathsf{hom}^2_B(f, g; h).$$

Proof. We may construct a map from left to right via path induction: in the case that h is $g \circ f$, we have the canonical witness of composition $\mathsf{comp}_{f,g}$. For equivalence, we translate to a map of total spaces in which case both sides become contractible. $\square$

Adjunctions In 1-category theory, we have two common and equivalent ways of defining adjunctions. Given functors $L : A \rightleftarrows B : R$, we may define them to be an adjoint pair $L \dashv R$ if they induce natural bijections on hom-sets

$$\mathrm{Hom}_B(La, b) \cong_{\mathbf{Set}} \mathrm{Hom}_A(a, Rb)$$

for all a and b. Let us call this a **transposing adjunction**, because of their symbolic analogy to transpose linear operators.

Alternatively, one may define adjunctions in terms of a *unit* $\eta : \mathrm{id}_A \Rightarrow RL$ and a *counit* $\epsilon : LR \Rightarrow \mathrm{id}_B$, such that we have higher order coherences

$$\begin{tikzcd} & RLR \arrow[dr, "R\epsilon"] & \\ R \arrow[ur, "\eta R"] \arrow[rr, equal] & & R \end{tikzcd} \quad \text{and} \quad \begin{tikzcd} & LRL \arrow[dr, "\epsilon L"] & \\ L \arrow[ur, "L\eta"] \arrow[rr, equal] & & L \end{tikzcd}$$

We call this notion of adjunction a **diagramatic adjunction**, due to its use of diagrammatic coherences.

When one attempts to formulate adjunctions for higher categories, these notions fail to be equivalent, however, and issues similar to the ones for equivalences replay themselves. In essence, this manifests itself in the fact that the coherences of a diagrammatic adjunction may fail to satisfy even higher-order coherences. Hence its formulation in type theory we dub **quasi-diagramatic adjunction**.

This situation may be redeemed in multiple ways as treated by Riehl and Shulman based on earlier work by Riehl and Verity [RV16]. One method is by appending additional coherences until higher homotopies are killed off. A minimal such example includes one additional two-dimensional coherence and two three-dimensional coherences, yielding the notion of **half-adjoint adjunctions**.

Another way is by separating the two roles played by either the unit or the counit. This yields **bi-diagramatic adjunctions**, in which one asks for a left and a right (co)unit, each satisfying one of the coherence laws stated earlier.

However, it turns out that the naive formulation of transposing adjunctions transfers without issues, as long as we use a coherent notion of equivalence. And so to keep things simple, this is the notion we will have in mind when speaking of adjunctions.

Definition 15.13 (Transposing adjunction)**.** The maps between types $L : A \rightleftarrows B : R$ define a **transposing adjunction** between A and B, where L **is left adjoint to** R, and R **is right adjoint to** L, written $L \dashv R$, if we have a family of equivalences of hom-types

$$\mathsf{T}_{L,R} : \prod_{a:A, b:B} (\mathsf{hom}_B(La, b) \simeq \mathsf{hom}_A(a, Rb)) .$$

In particular, from a transposing adjunction we may recover the unit and counit componentwise as

$$\eta_a = \mathsf{T}(a, La, \mathsf{id}_a) \quad \text{and} \quad \epsilon_b = \mathsf{T}^{-1}(Rb, b, \mathsf{id}_b).$$

15.2 Isomorphisms and Rezk types

We define the type of isomorphisms exactly analogously to equivalences of types. Again we use the notion of bi-invertibility to ensure homotopy uniqueness of isomorphism proofs for Segal types [RS17, Proposition 10.2].

Definition 15.14 (Isomorphism)**.** Given an arrow $f : x \to_A y$ in any type A, we define it to be an isomorphism if it is *bi-invertible*:

$$\mathsf{isIso}\, f :\equiv \left(\sum_{g : y \to_A x} \mathsf{hom}^2_A(g, f; \mathsf{id}_y) \right) \times \left(\sum_{h : y \to_A x} \mathsf{hom}^2_A(f, h; \mathsf{id}_x) \right).$$

We define the type of isomorphisms between two terms as

$$(x \cong_A y) :\equiv \sum_{f : x \to_A y} \mathsf{isIso}\, f.$$

Just as we saw in Section 5.1 that the interval type $\mathtt{I}$ covariantly represents the type of paths in a type, we may characterize the type of isomorphisms with a representing object:

Construction 15.15 (The free-standing bi-invertible arrow, [BW21, 4.2.1])**.** The free-standing bi-invertible arrow type $\mathbb{E}$ may be formally defined as the colimit of the following diagram of shapes

$$\begin{tikzcd}
& \Delta^1 \arrow[dl] \arrow[dr, "d^1"] & & \Delta^1 \arrow[dl, "d^0"'] \arrow[dr, "d^2"] & & \Delta^1 \arrow[dl, "d^1"'] \arrow[dr] & \\
\Delta^0 & & \Delta^2 & & \Delta^2 & & \Delta^0.
\end{tikzcd}$$

Hence it may be depicted as the following simplicial type with fillers

$$\begin{tikzcd}
& \cdot \arrow[r, equal] \arrow[dr] & \cdot \\
\cdot \arrow[r, equal] \arrow[ur] & \cdot \arrow[ur] &
\end{tikzcd}$$

This type covariantly represents the type of isomorphisms

$$\left(\sum_{x, y : A} x \cong_A y \right) \simeq (\mathbb{E} \to B).$$

Proposition 15.16. *Given a morphism $f : x \to_B y$ in a Segal type B, then the covariant and contravariant composition operations induced by f give equivalences of hom-types for all z if and only if f is an isomorphism*

$$(z \to_B x) \overset{f\circ -}{\simeq} (z \to_B y) \quad \textit{and} \quad (y \to_B z) \overset{-\circ f}{\simeq} (x \to_B z).$$

Proof. By duality, it suffices to consider the covariant case. Assume f is an isomorphism and let g be its right-inverse, then we claim that $g \circ -$ is a right-inverse of $f \circ -$. So let us verify this, using Proposition 15.11 at the second step:

$$(f \circ -) \circ (g \circ -) \equiv (f \circ (g \circ -)) \overset{\text{assoc}}{=} ((f \circ g) \circ -) \overset{\text{r-inv}}{=} (\mathsf{id}_y \circ -) \overset{\text{l-unit}}{=} \mathrm{id}$$

The argument is similar for the left-inverse of f.

Conversely, assume that $f \circ -$ is an equivalence for all z and let r be its right-inverse. Then in particular we have an equivalence $(y \to_B x) \simeq (y \to_B y)$ from which we may define an arrow $g :\equiv r(\mathsf{id}_y) : y \to_B x$ which by the following computation is a right-inverse of f:

$$f \circ g \equiv f \circ (r(\mathsf{id}_y)) \overset{\Pi\text{-comp}}{\equiv} (f \circ -)(r(\mathsf{id}_y)) \overset{\Pi\text{-comp}}{\equiv} ((f \circ -) \circ r)(\mathsf{id}_y) \overset{\text{r-inv}}{=} \mathsf{id}_y\,.$$

The argument is similar for the left-inverse. ☐

Definition 15.17 (Rezk type)**.** A Segal type B is a **Rezk type** if the canonical comparison map

$$\mathsf{id\text{-}to\text{-}iso}_B : \prod_{x,y:B} ((x =_B y) \to (x \cong_B y)) \qquad \mathsf{id\text{-}to\text{-}iso}_B(\mathsf{refl}_x) :\equiv \mathsf{id}_x$$

is an equivalence,

$$\mathsf{isRezk}\, B :\equiv \mathsf{isSegal}\, B \times \mathsf{isEquiv}(\mathsf{id\text{-}to\text{-}iso}_B).$$

Remark 15.18. Analogously to the definition of Rezk spaces in Definition 12.12, Rezk types may equivalently be characterized as Segal types which are local with respect to the terminal projection $\mathbb{E} \to \mathbf{1}$. Conversely, Rezk spaces may be characterized by an analogous map $X_0 \to X_1^{\sim}$ assigning each vertex x to its identity morphism in the subspace of X_1 generated by the weak equivalences [Rez01, Section 6].

We would also like to characterize the types which are only categories in a trivial sense, mimicking the relationship between sets and 1-categories in classical mathematics. These are types whose types of arrows and types of paths coincide, at least up to equivalence. Hence, the following natural definition follows:

Definition 15.19 (Discrete type)**.** A type A is a **discrete type** if the canonical comparison map

$$\mathsf{id\text{-}to\text{-}arr}_A : \prod_{x,y:A} ((x =_A y) \to (x \to_A y)) \qquad \mathsf{id\text{-}to\text{-}arr}_A(\mathsf{refl}_x) :\equiv \mathsf{id}_x$$

is an equivalence,

$$\mathsf{isDiscrete}\, A :\equiv \mathsf{isEquiv}(\mathsf{id\text{-}to\text{-}arr}_A).$$

Proposition 15.20 ([RS17, Proposition 10.10])**.** *Discrete types are precisely Rezk types in which all arrows are invertible.*

Proof. Observe that $\mathsf{id\text{-}to\text{-}arr}$ may be factored as the composite

$$(x =_A y) \xrightarrow{\mathsf{id\text{-}to\text{-}iso}} (x \cong_A y) \hookrightarrow (x \to_A y),$$

Since being an isomorphism is a proposition, the inclusion $(x \cong_A y) \hookrightarrow (x \to_A y)$ is an embedding. And so by [UF13, Theorem 4.6.3] it is an equivalence iff it is a surjection. Now, being Rezk and having all invertible arrows implies that both factors are equivalences, and so $\mathsf{id\text{-}to\text{-}arr}$ is a composite of equivalences and hence an equivalence itself. Conversely, if $\mathsf{id\text{-}to\text{-}arr}$ is an equivalence, then in particular its second factor is surjective. Hence by the 2-out-of-3 property for equivalences, $\mathsf{id\text{-}to\text{-}iso}$ is an equivalence as well. □

This means that the discrete types are the internal ∞-groupoids.

Proposition 15.21 ([RS17, Proposition 8.29])**.** *Given a Segal type B, then for any $b, b' : B$, the hom-type $\mathsf{hom}_B(b, b')$ is discrete.*

In fact, even more is true. The type family $\mathsf{hom}_B : B \to B \to \mathcal{U}$ is a two-sided discrete fibration, meaning it varies contravariantly over its first argument, and covariantly over its second argument.

Proof. See [RS17, Proposition 8.29]. □

15.3 Some categorical constructions

Let us now visit a couple of common categorical constructions. We discuss them here to continue the theme of categorical constructions, although we will soon find use for them all. We go into some detail with each construction, but no surprises will show up. It is worth reiterating that since our type theory is homotopy invariant, every universal property is weak in the sense that composition and uniqueness is always up to homotopy.

To begin we consider pullbacks of types. In our type theory, as with homotopy type theory, every pullback of types exists and admits an explicit internal description in terms of Σ- and identity types.

Theorem 15.22 (Pullback of types)
Given a cospan of types $f : B \to A \leftarrow C : g$, the type

$$S :\equiv \sum_{b:B, c:C} (f(b) = g(c)) \qquad \begin{tikzcd} S \arrow[r, "\mathsf{pr}_1"] \arrow[d, "\mathsf{pr}_0"'] \arrow[dr, phantom, "\lrcorner", very near start] & C \arrow[d, "g"] \\ B \arrow[r, "f"'] & A. \end{tikzcd}$$

satisfies the universal property of being the pullback of this cospan among both types and shapes. By this we mean that for any type or shape X, we have an equivalence

$$(X \to S) \simeq \left(\sum_{\substack{b:X\to B,\\ c:X\to C}} f \circ b = g \circ c \right)$$

Proof. Depending on whether X is a type or a shape, we apply either the type-theoretic axiom of choice for types twice or the type-theoretic axiom of relative choice for shapes on the inclusion $\varnothing \subseteq X$ twice to get the equivalences

$$\left(X \to \sum_{b:B,c:C} f(b) = g(c)\right) \simeq \left(\sum_{(b:X\to B)} \prod_{(x:X)} \sum_{(c:C)} f(b(x)) = g(c)\right) \simeq \left(\sum_{\substack{b:X\to B,\\ c:X\to C}} \prod_{(x:X)} f(b(x)) = g(c(x))\right)$$

at which point we may apply function extensionality if X is a type, or weak relative function extensionality on the inclusion $\varnothing \subseteq X$ if X is a shape

$$\left(\sum_{\substack{b:X\to B,\\ c:X\to C}} \prod_{(x:X)} f(b(x)) = g(c(x))\right) \simeq \left(\sum_{\substack{b:X\to B,\\ c:X\to C}} f \circ b = g \circ c\right). \qquad \square$$

Note that while we apply the function extensionality axioms in this proof, they are not strictly necessary for the Σ-type to satisfy an appropriate universal property. Without function extensionality, however, we would need to weaken our notion of uniqueness to that of pointwise uniqueness.

Example 15.23 (Type of fiberwise maps)**.** Given a map or shape inclusion $j : Y \to X$ and a map $\pi : E \to B$ we define the **type of fiberwise maps** between j and π, $j \rightrightarrows \pi$. This type consists of commutative square whose left leg is j and right leg is π. It may also be characterized by the following pullback

$$j \rightrightarrows \pi :\equiv \sum_{\substack{f:Y\to E,\\ g:X\to B}} \pi \circ f = g \circ j \quad \text{or} \quad \sum_{f:Y\to E} \left\langle X \to B\big|^{Y}_{\pi\circ f}\right\rangle \qquad \begin{tikzcd} j \rightrightarrows \pi \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & E^Y \arrow[d, "\pi^Y"] \\ B^X \arrow[r, "B^j"'] & B^Y. \end{tikzcd}$$

A note on terminology here, to us a *fiberwise map* consists of both a map of total spaces and a map of base spaces. This is in contrast to *fibered maps*, which stay constant at the base.

Warning 15.24. Some care must be taken when we conflate shapes and types in this way. Firstly, there are three cases to consider when j is a map or a shape inclusion since if j is a map it may either be a map from a type to another type or it may be a map from a shape to a type. Due to our abuse of notation, these types are denoted equivalently. Secondly, the strictness of identities varies among these. However, thanks to the results of Section 14.1, we have many analogous results between shapes and types which makes treatments of these conflations smoother.

Definition 15.25 (Leibniz cotensor)**.** Given maps $j : Y \to X$ and $\pi : E \to B$ as just discussed, i.e. j may be an inclusion of shapes, the **Leibniz cotensor** or **Leibniz**

exponential or **pullback hom** $j \mathbin{\hat{\pitchfork}} \pi$ is the canonically induced map in the following diagram

$$\begin{tikzcd}
E^X \arrow[rr, "E^j"] \arrow[dr, dashed, "j\hat{\pitchfork}\pi"] \arrow[dd, "\pi^X"'] & & E^Y \arrow[dd, "\pi^Y"] \\
& j \rightrightarrows \pi \arrow[ur] \arrow[dl] \arrow[dr, phantom, "\lrcorner", very near start] & \\
B^X \arrow[rr, "B^j"'] & & B^Y.
\end{tikzcd}$$

This map can be interpreted as taking lifts between j and π to the lifting problem they solve

$$\left\langle \begin{array}{ccc} Y & & E \\ j\downarrow & \nearrow & \downarrow\pi \\ X & & B \end{array} \right\rangle \xrightarrow{j\hat{\pitchfork}\pi} \left\langle \begin{array}{ccc} Y & \dashrightarrow & E \\ j\downarrow & & \downarrow\pi \\ X & \dashrightarrow & B \end{array} \right\rangle.$$

This interpretation allows us to give a second characterization of lifting properties which is more algebraic in flavour. A proof of a lifting property between j and π corresponds to a section of the Leibniz cotensor, while orthogonality corresponds to $j \mathbin{\hat{\pitchfork}} \pi$ being an equivalence.

In the degenerate case that π is the unique map to the terminal type $! : E \to \mathbf{1}$, the type of fiberwise maps $j \rightrightarrows\, !$ degenerates to E^Y and the Leibniz cotensor to E^j via the equivalences $\mathbf{1}^X \simeq \mathbf{1} \simeq \mathbf{1}^Y$.

There is much more that can be said about Leibniz cotensors. However, we focus on a single property that we need for a later argument.

Proposition 15.26 (Composition factorization property of leibniz cotensors)**.** *Given a map $\pi : E \to B$ and two composable maps $Z \xrightarrow{i} Y \xrightarrow{j} X$ where i and j may be shape inclusions, then $(j \circ i) \mathbin{\hat{\pitchfork}} \pi$ factorizes as $j \mathbin{\hat{\pitchfork}} \pi$ followed by a pullback of $i \mathbin{\hat{\pitchfork}} \pi$,*

$$(j \circ i) \mathbin{\hat{\pitchfork}} \pi = (i \mathbin{\hat{\pitchfork}} \pi)^{\lrcorner} \circ (j \mathbin{\hat{\pitchfork}} \pi).$$

Proof. We begin by constructing a big commutative diagram, enumerating two squares for further reference,

$$\begin{tikzcd}
E^X \arrow[d, dashed, "j\hat{\pitchfork}\pi"] \arrow[dr, "E^j"] \arrow[dd, bend right=60, "(j\circ i)\hat{\pitchfork}\pi"'] & & \\
j \rightrightarrows \pi \arrow[r] \arrow[d, dashed] \arrow[dr, phantom, "\text{(ii)}"] & E^Y \arrow[d, dashed, "i\hat{\pitchfork}\pi"] \arrow[dr, "E^i"] & \\
(j \circ i) \rightrightarrows \pi \arrow[r, dashed] \arrow[d] \arrow[dr, phantom, "\text{(i)}"] & i \rightrightarrows \pi \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & E^Z \arrow[d, "\pi^Z"] \\
B^X \arrow[r, "B^j"'] & B^Y \arrow[r, "B^i"'] & B^Z.
\end{tikzcd}$$

The dashed arrows are induced by the commutative squares

$$\begin{tikzcd} (j \circ i) \rightrightarrows \pi \arrow[r] \arrow[d] & E^Z \arrow[d] \\ B^X \arrow[r] & B^Z \end{tikzcd} \quad \text{and} \quad \begin{tikzcd} j \rightrightarrows \pi \arrow[r] \arrow[d] & E^Y \arrow[d] \\ B^X \arrow[r] & B^Y. \end{tikzcd}$$

Moreover, since these are pullback squares we get by left-cancelation that (i) and then (ii) are pullback squares. Hence we have obtained the desired factorization of $(j \circ i) \mathbin{\hat{\pitchfork}} \pi$. $\square$

Definition 15.27 (Comma type)**.** Given a cospan of types $f : B \to A \leftarrow C : g$, the **comma type** or **comma construction** $f \downarrow g$ is defined as the pullback

$$\begin{tikzcd} f \downarrow g \arrow[r, "\mu"] \arrow[d, "(\mathsf{pr}_0, \mathsf{pr}_1)"'] \arrow[dr, phantom, "\lrcorner", very near start] & A^{\Delta^1} \arrow[d, "(\mathsf{pr}_0, \mathsf{pr}_1)"] \\ B \times C \arrow[r, "f \times g"] & A^{\partial \Delta^1} \end{tikzcd} \quad \text{or the terminal cone of} \quad \begin{tikzcd} B \arrow[d, "f"'] & f \downarrow g \arrow[l, dashed, "\mathsf{pr}_0"'] \arrow[r, dashed, "\mathsf{pr}_1"] \arrow[d, dashed] & C \arrow[d, "g"] \\ A & A^{\Delta^1} \arrow[l, "\mathsf{pr}_0"'] \arrow[r, "\mathsf{pr}_1"] & A. \end{tikzcd}$$

The comma type $f \downarrow g$ admits a particularly simple internal description thanks to the strictness of extension types

$$f \downarrow g \simeq \sum_{b:B, c:C} (f(b) \to_A g(c)).$$

As we can see clearly from this description, the comma type $f \downarrow g$ consists of arrows in A whose domain and codomain are parameterized by the functions f and g. The comma type also comes equipped with a canonical natural transformation μ given by the pullback of $f \times g$, called the **comma cone** of $f \downarrow g$:

$$\begin{tikzcd} & f \downarrow g \arrow[dl, "\mathsf{pr}_0"'] \arrow[dr, "\mathsf{pr}_1"] & \\ B \arrow[rr, phantom, "\overset{\mu}{\Longrightarrow}"] \arrow[dr, "f"'] & & C \arrow[dl, "g"] \\ & A. & \end{tikzcd}$$

The comma types $f \downarrow \mathrm{id}_A$ and $\mathrm{id}_A \downarrow g$, usually denoted $f \downarrow A$ and $A \downarrow g$, admit particularly simple diagrammatic descriptions as the following:

$$\begin{tikzcd} & f \downarrow A \arrow[dl, "\mathsf{pr}_0"'] \arrow[dr, "\mathsf{pr}_1"] \arrow[d, phantom, "\Longrightarrow"] & \\ B \arrow[rr, "f"'] & {} & A \end{tikzcd} \qquad \begin{tikzcd} & A \downarrow g \arrow[dl, "\mathsf{pr}_0"'] \arrow[dr, "\mathsf{pr}_1"] \arrow[d, phantom, "\Longrightarrow"] & \\ A & {} & C. \arrow[ll, "g"] \end{tikzcd}$$

As a special case of comma constructions we recover the (co)slice constructions:

Definition 15.28 ((Co)slice type, [RV21, Warning 4.2.10])**.** Let A be a type and Δ be the diagonal map $\Delta : A \to A^J$. Given an X-indexed family of J-shaped diagrams in A, $\psi : X \to A^J$, we define the **coslice type** ψ/A as the comma construction

$$\psi/A :\equiv \psi \downarrow \Delta \quad \text{of the cospan} \quad X \overset{\psi}{\to} A^J \overset{\Delta}{\leftarrow} A.$$

Dually, the slice type is $A/\psi :\equiv \Delta \downarrow \psi$. In particular, given a term $a : A$ we have the over and under types $a/A \simeq a \downarrow A$ and $A/a \simeq A \downarrow a$.

$$\psi/A \simeq \sum_{x:X, a:A} \psi(x) \Rightarrow \Delta(a) \simeq \sum_{(x:X, a:A)} \prod_{(j:J)} (\psi(x,j) \to_A a)$$

$$\begin{tikzcd} \psi/A \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & \left(A^J\right)^{\Delta^1} \arrow[d] \\ X \times A \arrow[r, "\psi \times \Delta"] & A^J \times A^J \end{tikzcd}$$

Given a cospan of types $f : B \to A \leftarrow C : g$ the pullback property of $f \downarrow g$ states that to construct a unique map $X \to f \downarrow g$ it suffices to provide maps $b : X \to B$, $c : X \to C$ and $\alpha : X \to A^{\Delta^1}$ such that $\mathsf{pr}_0^X \alpha = f \circ b$ and $\mathsf{pr}_1^X \alpha = g \circ c$. Interpreting α as a natural transformation yields a second formulation of the pullback property of the comma type.

Remark 15.29 (Universal property of comma type)**.** Given a natural transformation over the cospan of types $f : B \to A \leftarrow C : g$ as below-left, then there exists a unique map $a : X \to f \downarrow g$ such that the below-right diagram commutes

$$\begin{tikzcd} & X \arrow[dl, "b"'] \arrow[dr, "c"] & \\ B \arrow[dr, "f"'] \arrow[rr, phantom, "\overset{\alpha}{\Longrightarrow}"] & & C \arrow[dl, "g"] \\ & A & \end{tikzcd} \quad = \quad \begin{tikzcd} & X \arrow[dl, "b"'] \arrow[dr, "c"] \arrow[d, dashed, "a"] & \\ B \arrow[dr, "f"'] & f \downarrow g \arrow[l, "\mathsf{pr}_0"'] \arrow[r, "\mathsf{pr}_1"] \arrow[d, phantom, "\underset{\mu}{\Longrightarrow}"] & C \arrow[dl, "g"] \\ & A & \end{tikzcd}$$

and moreover, whiskering with the comma cone μ gives an inverse to this correspondence.

16 Fibrations and families

Moving to the context of fibrations and families, we again have use of the existence of a universe type and the straightening-unstraightening construction as in Section 6. This may be done precisely as before, although now we also require the universe to be closed under the formation of extension types.

Using the straightening-unstraightening equivalence we may treat fibrations and type families interchangably. Accordingly, given a predicate on families $\mathcal{P} : \mathsf{Fam}\, \mathcal{U} \to \mathsf{Prop}\, \mathcal{U}$, we may say that the map $\pi : E \to B$ satisfies $\mathcal{P}$ if its straightening does, and conversely. In this vein, given families $P : B \to \mathcal{U}$ and $Q : (\sum_{b:B} P(b)) \to \mathcal{U}$ we define their composite family

$$Q \circ P :\equiv b \mapsto \left(\sum_{e:P(b)} Q(b,e) \right) : B \to \mathcal{U}.$$

When working with type families, we may define types of dependent arrows.

Definition 16.1 (Dependent arrow)**.** Suppose given a type family $P : B \to \mathcal{U}$ and an arrow in the base space $u : b \to_B b'$. Then given terms e and e' lying over b and b' respectively, we define the type of **dependent arrows**

$$\mathsf{hom}^P_u(e, e') :\equiv \left\langle \prod_{t:\Delta^1} P(u(t)) \middle|^{\partial\Delta^1}_{[b,b']} \right\rangle.$$

And of course we can repeat the constructions like n-cells, compositions and so on in the dependent case. From now on, we will take these constructions for granted, and trust that the reader may find the right definitions by themselves if need be.

Of course dependent arrows also correspond precisely to non-dependent arrows in the total space of a type family.

16.1 The lifting problem

In this section, we revisit the idea of lifting problems as was done in the context of model categories in Section 9, but now in the context of the type theory. As we will see, lifting properties are efficient abstractions for many different concepts in the theory. Moreover, with the help of the Leibniz cotensor as defined in the previous section, we have a second way of characterizing lifting properties which lends itself well to the current discussion.

Definition 16.2 (Lifting property)**.** Given a function or shape inclusion $j : Y \to X$ and a function $\pi : E \to B$ we say j has the left lifting property with respect to π and π has the right lifting property with respect to j, denoted $j \boxslash \pi$, if any commutative square as follows admits a lifting map as marked with a dashed arrow preserving commutativity

$$\begin{tikzcd} Y \arrow[r] \arrow[d, "j"'] & E \arrow[d, "\pi"] \\ X \arrow[r] \arrow[ur, dashed] & B. \end{tikzcd}$$

If such liftings are unique the morphisms are said to be **orthogonal**, written $j \perp \pi$. In addition, we will say that π is j-orthogonal. We will say that a type or shape X has the left lifting property with respect to π if the unique inclusion of the empty type or shape $\mathbf{0} \hookrightarrow X$ has this left lifting property, and we will say E has the right lifting property with respect to j if the unique map $E \to \mathbf{1}$ does. We define the types

$$j \boxslash \pi :\equiv \prod_{\substack{f:Y\to E,\\ g:X\to B}} \left((\pi \circ f = g \circ j) \to \sum_{h:X\to E} (h \circ j = f) \times (\pi \circ h = g) \right)$$

$$j \perp \pi :\equiv \prod_{\substack{f:Y\to E,\\ g:X\to B}} \left((\pi \circ f = g \circ j) \to \mathsf{isContr}\left(\sum_{h:X\to E} (h \circ j = f) \times (\pi \circ h = g) \right) \right)$$

or if j is a shape inclusion

$$
\begin{aligned}
j \boxslash \pi &:\equiv \prod_{f:Y \to E} \left(\prod_{g:\left\langle X \to B \middle|^{Y}_{\pi \circ f} \right\rangle} \sum_{h:\left\langle X \to E \middle|^{Y}_{f} \right\rangle} \pi \circ h = g \right) \\
j \perp \pi &:\equiv \prod_{f:Y \to E} \left(\prod_{g:\left\langle X \to B \middle|^{Y}_{\pi \circ f} \right\rangle} \mathsf{isContr} \left(\sum_{h:\left\langle X \to E \middle|^{Y}_{f} \right\rangle} \pi \circ h = g \right) \right).
\end{aligned}
$$

Lemma 16.3. *The morphism j lifts against π if and only if $j \mathbin{\hat{\pitchfork}} \pi$ admits a section. Moreover j is orthogonal to π if and only if $j \mathbin{\hat{\pitchfork}} \pi$ is an equivalence.*

Proof. This is just a reformulation of their defining properties, noting that we always have a map in the opposite direction given by precomposing with j and postcomposing with π. □

Corollary 16.4. *Orthogonality is a proposition.* □

Let us give a few examples/definitions in terms of lifting properties.

Example 16.5 (Examples of lifting properties)**.**

(i) A type is Segal iff it is $(\Lambda^2_1 \hookrightarrow \Delta^2)$-orthogonal.

(ii) A type is Rezk iff it is Segal and $(\mathbb{E} \to \mathbf{1})$-orthogonal.

(iii) A type is discrete iff it is $(\Delta^1 \to \mathbf{1})$-orthogonal.[26]

(iv) A map is surjective iff it has the right lifting property with respect to $\mathbf{0} \hookrightarrow \mathbf{1}$.

(v) A map is bijective iff it is $(\mathbf{0} \hookrightarrow \mathbf{1})$-orthogonal.

(vi) A map is full iff it has the right lifting property with respect to $\partial\Delta^1 \hookrightarrow \Delta^1$.

(vii) A map is fully faithful iff it is $(\partial\Delta^1 \hookrightarrow \Delta^1)$-orthogonal.

(viii) A map is conservative iff it has the right lifting property with respect to the middle inclusion $\Delta^1 \hookrightarrow \mathbb{E}$.

(ix) A family is covariant iff it is orthogonal with respect to the initial vertex inclusion $i_0 : \mathbf{1} \hookrightarrow \Delta^1$ and contravariant iff it is orthogonal with respect to the terminal vertex inclusion $i_1 : \mathbf{1} \hookrightarrow \Delta^1$.

(x) A family is *inner* iff it is $(\Lambda^2_1 \hookrightarrow \Delta^2)$-orthogonal, generalizing Segal types.

[26] Although this is not a definable map, the example may still be translated to something meaningful by elaborating on the types involved.

(xi) A family is *isoinner* iff it is inner and $(\mathbb{E} \to \mathbf{1})$-orthogonal,[27] generalizing Rezk types.

The closure properties of left and right lifting properties as proven in Section 9 still apply with analogous proofs in our current setting. Let us however prove a couple of closedness properties for orthogonality.

Proposition 16.6 (Closed under dependent product, [BW21, Proposition 3.1.4])**.** *Given a map $j : Y \to X$ and a family of fibrations $\pi_{(-)} : I \to \mathsf{Fib}\,\mathcal{U}$ for some indexing type or shape I such that for each $i : I$ the fibration π_i is j-orthogonal, then the dependent product $\prod_{i:I} \pi_i$ is j-orthogonal.*

Proof. By hypothesis we have pullback-squares

$$\begin{tikzcd} E_i^Y \arrow[r, "E_i^j"] \arrow[d, "\pi_i^Y"'] \arrow[dr, phantom, "\lrcorner", very near start] & E_i^X \arrow[d, "\pi_i^X"] \\ B_i^Y \arrow[r, "B_i^j"'] & B_i^X . \end{tikzcd}$$

It may be shown that taking dependent products preserves pullbacks

$$\begin{tikzcd} \prod_{i:I} E_i^Y \arrow[r, "\prod_{i:I} E_i^j"] \arrow[d, "\prod_{i:I} \pi_i^Y"'] \arrow[dr, phantom, "\lrcorner", very near start] & \prod_{i:I} E_i^X \arrow[d, "\prod_{i:I} \pi_i^X"] \\ \prod_{i:I} B_i^Y \arrow[r, "\prod_{i:I} B_i^j"'] & \prod_{i:I} B_i^X , \end{tikzcd}$$

and by Corollary 14.3, dependent products commute with exponents

$$\begin{tikzcd} (\prod_{i:I} E_i)^Y \arrow[r, "(\prod_{i:I} E_i)^j"] \arrow[d, "(\prod_{i:I} \pi_i)^Y"'] \arrow[dr, phantom, "\lrcorner", very near start] & (\prod_{i:I} E_i)^X \arrow[d, "(\prod_{i:I} \pi_i)^X"] \\ (\prod_{i:I} B_i)^Y \arrow[r, "(\prod_{i:I} B_i)^j"'] & (\prod_{i:I} B_i)^X . \end{tikzcd}$$

□

Proposition 16.7 (Closed under dependent sums, [BW21, Proposition 3.1.15])**.** *Let $j : Y \to X$ be a map or shape inclusion and $\pi : E \to B$ be a map such that B is j-orthogonal. Then E is j-orthogonal if and only if π is.*

Proof. We have the diagram

$$\begin{tikzcd} E^X \arrow[rr, "E^j"] \arrow[dr, "j \hat{\pitchfork} \pi"] \arrow[dd, "\pi^X"'] & & E^Y \arrow[dd, "\pi^Y"] \\ & j \rightrightarrows \pi \arrow[ur, "\simeq"] \arrow[dl] \arrow[dr, phantom, "\lrcorner", very near start] & \\ B^X \arrow[rr, "\simeq", "B^j"'] & & B^Y \end{tikzcd}$$

[27] This is compatible with the definition of Buchholtz and Weinberger thanks to [BW21, Corollary 4.2.5].

in which B^j and its pullback are equivalences since B is j-orthogonal. Now if π is j-orthogonal, then their leibniz cotensor $j \mathbin{\hat{\pitchfork}} \pi$ is an equivalence, and so E^j as a composition of equivalences is an equivalence as well. Conversely, if E is j-orthogonal, then E^j is an equivalence, and so $j \mathbin{\hat{\pitchfork}} \pi$ is an equivalence by the 2-out-of-3 property. $\square$

Remark 16.8. From these two propositions, we have that Segal types, Rezk types, and discrete types are all closed under dependent products, in particular binary products and exponents. Moreover, they are also closed under dependent sums over indexing-types with the same property. In particular, they are closed under binary sums.

LARI liftings. For some uses, right orthogonality is too strict and the right lifting property too lax. However, using the Leibniz cotensor construction we have a convenient framing for phrasing lifting properties.

Definition 16.9 (Left adjoint right inverse (LARI))**.** A map $L : A \to B$ is a **left adjoint right inverse** (LARI) if it is a left adjoint such that the unit of the adjunction is a natural isomorphism $\eta : RL \cong \mathrm{id}_A$. Hence it acts as a right inverse to its adjoint. Dually, we have the notions of LALI, RALI, and RARI maps. If L is a left adjoint right inverse, then its adjoint R is a right adjoint left inverse.

Remark 16.10. A left adjoint is LARI if and only if it is fully faithful. If it is LARI, we have the right-hand equivalence

$$\mathsf{hom}_B(L(a), L(a')) \overset{\mathrm{LA}}{\simeq} \mathsf{hom}_A(a, R(L(a'))) \overset{\mathrm{RI}}{\simeq} \mathsf{hom}_A(a, a'),$$

and conversely if it is fully faithful we have an equivalence of the left-most and right-most types.

Example 16.11. The diagonal map is left adjoint right inverse to the domain projection map

$$\begin{tikzcd} E \arrow[r, bend left, "\Delta"] \arrow[r, phantom, "\perp"] & E^{\Delta^1} \arrow[l, bend left, "\mathsf{pr}_0"] \end{tikzcd}.$$

The following notion serves as an intermediate step between the lifting property and orthogonality by applying the idea of LARI maps to the equivalence of Lemma 16.3:

Definition 16.12 (LARI lifting)**.** Given a map or shape inclusion $j : Y \to X$ and a fibration $\pi : E \to B$, then π is said to be j-LARI if the leibniz cotensor $j \mathbin{\hat{\pitchfork}} \pi$ has a left adjoint right inverse.

Cocartesian fibrations can for instance be characterized through the formalism of LARI lifting properties. This is the approach taken by Buchholtz and Weinberger [BW21].

16.2 Absolute lifting

In this section, we introduce the notion of an absolute left lifting diagram and prove a few basic properties for them. Of course, everyting in this section can be codualized to yield the notion of absolute right liftings.

Definition 16.13 (Absolute lifting diagram)**.** Given a cospan $X \xrightarrow{f} B \xleftarrow{\pi} E$ over a Segal type B then an **absolute left lifting of f along π** is a map $\ell : X \to E$ along with a natural transformation $\lambda : f \Rightarrow \pi\ell$ universal in the sense that any span $X \xleftarrow{j} Y \xrightarrow{h} E$ with a natural transformation $\alpha : fj \Rightarrow \pi h$ factorizes uniquely through λ:

$$\begin{tikzcd} Y \arrow[r, "h"] \arrow[d, "j"'] \arrow[dr, phantom, "\alpha\Uparrow"] & E \arrow[d, "\pi"] \\ X \arrow[r, "f"'] & B \end{tikzcd} \quad = \quad \begin{tikzcd} Y \arrow[r, "h"] \arrow[d, "j"'] \arrow[dr, phantom, "\beta\Uparrow!", pos=0.3] & E \arrow[d, "\pi"] \\ X \arrow[r, "f"'] \arrow[ur, "\ell"] \arrow[ur, phantom, "\Uparrow\lambda", pos=0.7, shift right=3] & B. \end{tikzcd}$$

We may define the type of proofs that (ℓ, λ) is an absolute left lifting as follows

$$\mathsf{isALL}_{\pi,f}(\ell,\lambda) :\equiv \prod_{(Y:\mathcal{U})} \prod_{(X \xleftarrow[j]{} Y \xrightarrow[h]{} E)} \prod_{(\alpha : fj \Rightarrow \pi h)} \mathsf{isContr}\left(\sum_{(\beta : \ell j \Rightarrow h)} \alpha = \pi\beta \circ \lambda j \right)$$

noting that the composites $\pi\beta \circ \lambda j$ are well-defined (in fact homotopy unique) since B is Segal. Moreover, we can see that isALL is a family of propositions since contractibility proofs always are [UF13, Lemma 3.11.4] and propositions are closed under dependent products [UF13, Example 3.6.2]. We define the type of absolute left liftings of f along π as the type

$$\mathsf{hasALL}(\pi, f) :\equiv \sum_{(\ell : X \to E)} \sum_{(\lambda : f \Rightarrow \pi\ell)} \mathsf{isALL}_{\pi,f}(\ell,\lambda).$$

An equivalent characterization of absolute left liftings is given by the following lemma.

Lemma 16.14. *A natural transformation $\lambda : f \Rightarrow \pi\ell$ with a Segal codomain is an absolute left lifting along π if and only if the following transposing map is an equivalence*

$$\beta \mapsto \pi\beta \circ \lambda j : \prod_{(Y:\mathcal{U})} \prod_{(X \xleftarrow[j]{} Y \xrightarrow[h]{} E)} \left((\ell j \Rightarrow h) \to (fj \Rightarrow \pi h) \right).$$

And in fact these types are equivalent propositions.

Proof. This is just a reformulation of the defining property. □

Corollary 16.15. *A natural transformation $\eta : \mathrm{id}_A \Rightarrow gf$ is the unit of an adjunction $f \dashv g$ if and only if (f, η) is an absolute left lifting of id_A along g.*

Lemma 16.16. *The type of absolute left liftings of f along π is a proposition when E is Rezk.*[28]

[28] It would in fact suffice to have the implication $\prod_{g,h:E^Y} \prod_{\alpha,\beta : g \Rightarrow h} (\alpha \cong \beta) \to (\alpha = \beta)$ and Segalness.

Proof. Assume given two absolute left liftings (ℓ, λ) and (ℓ', λ'). Then each factorizes through the other, meaning we have natural transformations $\chi : \ell \Rightarrow \ell'$ and $\chi' : \ell' \Rightarrow \ell$ such that $\lambda' = \pi\chi \circ \lambda$ and $\lambda = \pi\chi' \circ \lambda'$. In particular we have $\lambda = \pi\chi' \circ (\pi\chi \circ \lambda) = \pi(\chi' \circ \chi) \circ \lambda$, but another factorization of λ against itself is id_ℓ, hence by uniqueness of factorizations $\chi' \circ \chi = \mathsf{id}_\ell$, and similarly $\chi \circ \chi' = \mathsf{id}'_\ell$. So χ and χ' form a quasi-isomorphism[29] in E^Y, and so by Rezk completeness [RS17, Proposition 10.9] we have $\lambda = \lambda'$. □

Lemma 16.17 (Absolute liftings are absolute / closed under restriction)**.** *Given an absolute left lifting (ℓ, λ) of f along π and a function or shape inclusion $i : Z \to X$ into the domain of f, then $(\ell i, \lambda i)$ is an absolute left lifting of fi along π*

$$\mathsf{isALL}_{\pi,f}(\ell, \lambda) \to \mathsf{isALL}_{\pi,fi}(\ell i, \lambda i).$$

Proof. Given the lifting problem with boundary the outer square of the diagram

$$\begin{tikzcd}
Y \arrow[rr, "h"] \arrow[d, "j"'] \arrow[dr, "ij"] & \arrow[d, phantom, "\beta\Uparrow!"] & E \arrow[d, "\pi"] \\
Z \arrow[r, "i"'] \arrow[u, phantom, "||", near start] & X \arrow[r, "f"'] \arrow[ur, "\ell"] & B, \arrow[ul, phantom, "\Uparrow"]
\end{tikzcd}$$

then by the left lifting property of the original diagram we receive a natural transformation $\beta : \ell(ij) \Rightarrow h$ which by associativity solves our original lifting problem. Now assume given any other factorization $\beta' : (\ell i)j \Rightarrow h$, then it is clearly also a factorization of the lifting problem over λ, hence by contractibility it is equal to β. □

Remark 16.18. In particular, absolute left liftings are absolute left Kan lifts, meaning that they admit unique lifts of the spans $X = X \to E$.

Lemma 16.19 (Composition and left cancellation, [RV21, Lemma 2.4.1])**.** *Given a diagram as below where (ℓ', λ') is an absolute left lifting of ℓ along π',*

$$\begin{tikzcd}
F \arrow[r, "\pi'"] \arrow[dr, phantom, "\lambda'\Uparrow", near start] & E \arrow[d, "\pi"] \\
X \arrow[u, "\ell'"] \arrow[r, "f"'] \arrow[ur, "\ell"] & B \arrow[ul, phantom, "\Uparrow\lambda", near start]
\end{tikzcd}$$

then (ℓ, λ) is an absolute left lifting of f along π if and only if $(\ell', \pi\lambda' \circ \lambda)$ is an absolute left lifting of f along $\pi\pi'$.

Proof. Rereading the statement of the lemma we see that one direction means that absolute left liftings compose, while the other means that they cancel from the left.

So first let us prove that absolute left liftings compose. Assume (ℓ, λ) is an absolute lifting of f along π and that we are given the left-hand square as follows. The argument

[29] This is in the terminology of [UF13] and means that they inhabit the type of mutual inverses.

works best visually, hence we display the diagrams first, then follow up with a brief argument

$$\begin{tikzcd} Y \arrow[r, "h"] \arrow[dd, "j"'] & F \arrow[d, "\pi'"] \\ \arrow[r, phantom, "\alpha\Uparrow"] & E \arrow[d, "\pi"] \\ X \arrow[r, "f"'] & B \end{tikzcd} = \begin{tikzcd} Y \arrow[r, "h"] \arrow[dd, "j"'] & F \arrow[d, "\pi'"] \\ \arrow[r, phantom, "\alpha'\Uparrow"] & E \arrow[d, "\pi"] \\ X \arrow[r, "f"'] \arrow[ur, "\ell"] \arrow[r, phantom, shift left=2, "\lambda\Uparrow"] & B \end{tikzcd} = \begin{tikzcd} Y \arrow[r, "h"] \arrow[dd, "j"'] & F \arrow[d, "\pi'"] \\ \arrow[r, phantom, "\alpha''\Uparrow"] \arrow[r, phantom, shift right=3, "\lambda'\Uparrow"] & E \arrow[d, "\pi"] \\ X \arrow[r, "f"'] \arrow[ur, "\ell"] \arrow[uur, "\ell'"] \arrow[r, phantom, shift left=2, "\lambda\Uparrow"] & B. \end{tikzcd}$$

Given a natural transformation $\alpha : fj \Rightarrow \pi\pi'h$, we may factorize through (ℓ, λ) to obtain a unique natural transformation $\alpha' : \ell j \Rightarrow \pi'h$. This transformation may be uniquely factorized through (ℓ', λ') to get a natural transformation $\alpha'' : \ell'j \Rightarrow h$. Since this operation preserves commutativity of our diagram, it solves the originally posed factorization problem and it is unique.

Conversely, to show absolute left liftings cancel from the left, assume that $(\ell', \pi\lambda' \circ \lambda)$ is an absolute left lifting of f along $\pi\pi'$, then we want to prove that (ℓ, λ) is an absolute left lifting of f along π. So assume given the left-hand square as follows

$$\begin{tikzcd} Y \arrow[r, "h"] \arrow[dd, "j"'] & F \arrow[d, "\pi'"] \\ \arrow[r, phantom, "\alpha\Uparrow"] & E \arrow[d, "\pi"] \\ X \arrow[r, "f"'] & B \end{tikzcd} = \begin{tikzcd} Y \arrow[r, "h"] \arrow[dd, "j"'] & F \arrow[d, "\pi'"] \\ \arrow[r, phantom, shift left=3, "\alpha'\Uparrow"] \arrow[r, phantom, shift right=3, "\Uparrow \pi\lambda'\circ\lambda"] & E \arrow[d, "\pi"] \\ X \arrow[r, "f"'] \arrow[uur, "\ell'"] & B \end{tikzcd} = \begin{tikzcd} Y \arrow[r, "h"] \arrow[dd, "j"'] & F \arrow[d, "\pi'"] \\ \arrow[r, phantom, "\Uparrow \pi'\alpha'\circ\lambda'j"] & E \arrow[d, "\pi"] \\ X \arrow[r, "f"'] \arrow[ur, "\ell"] \arrow[r, phantom, shift left=2, "\Uparrow\lambda"] & B. \end{tikzcd}$$

The natural transformation α may be uniquely factorized through $\pi\lambda' \circ \lambda$ to get a natural transformation $\alpha' : \ell'j \Rightarrow h$. This we may compose with λ' to get a solution to the factorization problem.

For uniqueness, assume given another solution to the factorization problem $\beta : \ell j \Rightarrow \pi'h$ such that $\pi\beta \circ \lambda j = \alpha$. Then in particular its factorization through $\pi\lambda' \circ \lambda$ is equal to α', hence by commutativity $\beta = \pi'\alpha' \circ \lambda'j$. □

To end the section, we mention one other useful application of absolute lifting diagrams, their usage in defining limits.

Definition 16.20 (Colimit)**.** A **colimit** of an X-indexed family of diagrams $d : X \to A^J$ of shape J in A is an absolute left lifting diagram

$$\begin{tikzcd} & A \arrow[d, "\Delta"] \\ X \arrow[r, "d"'] \arrow[ur, "\operatorname{colim} d"] \arrow[r, phantom, shift left=3, "\Uparrow\eta"] & A^J. \end{tikzcd}$$

Example 16.21. An initial element of A may be defined as a colimit of the empty diagram

$$\begin{tikzcd} & A \arrow[d, "!"] \\ \mathbf{1} \arrow[r, equal] \arrow[ur, "\operatorname{colim} d"] \arrow[r, phantom, shift left=3, "\shortparallel"] & \mathbf{1}. \end{tikzcd}$$

Equivalently, initial elements of A may be identified via the following contractibility criterion: An element $a : A$ is **initial** if for all $x : A$, $\mathsf{hom}_A(a, x)$ is contractible.

16.3 Comma representability

Definition 16.22 (Comma representability)**.** Given a cospan $f : B \to A \leftarrow C : g$ of types, the comma type $f \downarrow g$ is **left representable** if there exists a map $\ell : B \to C$ such that the following comma types are fibered equivalent

$$f \downarrow g \simeq_{B \times C} \ell \downarrow C.$$

As you might've expected, comma representable types have a tight connection with absolute liftings.

Theorem 16.23 ([RV21, Theorem 3.5.3 codual])
Given a Segal type A, the triangle

$$\begin{tikzcd}
 & C \arrow[d, "g"] \\
B \arrow[ur, "\ell"] \arrow[r, "f"'] \arrow[r, phantom, shift left=2ex, "\Uparrow\lambda"] & A
\end{tikzcd}$$

defines an absolute left lifting diagram if and only if the following dashed map y induced by the universal property of the comma type is a fibered equivalence $y : \ell \downarrow C \simeq_{B \times C} f \downarrow g$

$$\begin{tikzcd}
 & \ell \downarrow C \arrow[dl, "\mathsf{pr}_0"'] \arrow[dr, "\mathsf{pr}_1"] \arrow[d, phantom, "\overset{\eta}{\Longrightarrow}"] & \\
B \arrow[rr, "\ell"] \arrow[dr, "f"'] & {} \arrow[d, phantom, "\overset{\lambda}{\Longrightarrow}"] & C \arrow[dl, "g"] \\
 & A &
\end{tikzcd}
=
\begin{tikzcd}
 & \ell \downarrow C \arrow[dl, "\mathsf{pr}_0"'] \arrow[dr, "\mathsf{pr}_1"] \arrow[d, dashed, "y"] & \\
B \arrow[dr, "f"'] & f \downarrow g \arrow[l, "\mathsf{pr}_0"'] \arrow[r, "\mathsf{pr}_1"] \arrow[d, phantom, "\overset{\mu}{\Longrightarrow}"] & C . \arrow[dl, "g"] \\
 & A &
\end{tikzcd}$$

Proof adapted. Suppose that (ℓ, λ) is an absolute left lifting of f along g, and consider the unique factorization μ' of the comma cone μ of $f \downarrow g$ through λ as depicted below center-left. By the universal property of comma types (15.29), there is a unique map $z : f \downarrow g \to \ell \downarrow C$ such that $\mathsf{pr}_i \circ z = \mathsf{pr}_i$ as depicted below center-right. Now by substituting in the commutative diagram for y, we see that $y \circ z : f \downarrow g \to f \downarrow g$ is a map that factors the comma cone for $f \downarrow g$ through itself. But $\mathrm{id}_{f \downarrow g}$ is also, so by the universal property of comma types we have $y \circ z = \mathrm{id}_{f \downarrow g}$.

$$\begin{tikzcd}
 & f \downarrow g \arrow[dl] \arrow[dr] & \\
B \arrow[rr, phantom, "\overset{\mu}{\Longrightarrow}"] \arrow[dr, "f"'] & & C \arrow[dl, "g"] \\
 & A &
\end{tikzcd}
=
\begin{tikzcd}
 & f \downarrow g \arrow[dl] \arrow[dr] \arrow[d, phantom, "\overset{\mu'}{\Longrightarrow}"] & \\
B \arrow[rr, "\ell"] \arrow[dr, "f"'] & {} \arrow[d, phantom, "\overset{\lambda}{\Longrightarrow}"] & C \arrow[dl, "g"] \\
 & A &
\end{tikzcd}
=
\begin{tikzcd}
 & f \downarrow g \arrow[d, dashed, "z"] \arrow[ddl, bend right] \arrow[ddr, bend left] & \\
 & \ell \downarrow C \arrow[dl] \arrow[dr] \arrow[d, phantom, "\overset{\eta}{\Longrightarrow}"] & \\
B \arrow[rr, "\ell"] \arrow[dr, "f"'] & {} \arrow[d, phantom, "\overset{\lambda}{\Longrightarrow}"] & C \arrow[dl, "g"] \\
 & A &
\end{tikzcd}
=
\begin{tikzcd}
 & f \downarrow g \arrow[d, dashed, "z"] \arrow[ddl, bend right] \arrow[ddr, bend left] & \\
 & \ell \downarrow C \arrow[dl] \arrow[dr] \arrow[d, dashed, "y"] & \\
B \arrow[dr, "f"'] & f \downarrow g \arrow[l] \arrow[r] \arrow[d, phantom, "\overset{\mu}{\Longrightarrow}"] & C . \arrow[dl, "g"] \\
 & A &
\end{tikzcd}$$

To show $z \circ y = \mathrm{id}_{\ell\downarrow C}$ we may argue similarly that the comma cone of ℓ restricts along $z \circ y$ to itself. Since (ℓ, λ) is an absolute left lifting it suffices to verify $\eta y z = \eta$. To verify this we start by pasting λ to get the below leftmost diagram. whiskering η with z we get the left-center diagram, and composing λ and μ' we get the right-center one. Now, the final equality is by the construction of y.

$$
\begin{tikzcd}
& \ell\downarrow C \arrow[d, dashed, "y"] \arrow[dddl, bend right] \arrow[dddr, bend left] & \\
& f\downarrow g \arrow[d, dashed, "z"] \arrow[ddl, bend right] \arrow[ddr, bend left] & \\
& \ell\downarrow C \arrow[dl] \arrow[dr] \arrow[d, phantom, "\Rightarrow\eta"] & \\
B \arrow[rr, "\ell"] \arrow[dr, "f"'] & {} \arrow[d, phantom, "\Rightarrow\lambda"] & C \arrow[dl, "g"] \\
& A &
\end{tikzcd}
=
\begin{tikzcd}
& \ell\downarrow C \arrow[d, dashed, "y"] \arrow[ddl, bend right] \arrow[ddr, bend left] & \\
& f\downarrow g \arrow[dl] \arrow[dr] \arrow[d, phantom, "\Rightarrow\mu'"] & \\
B \arrow[rr, "\ell"] \arrow[dr, "f"'] & {} \arrow[d, phantom, "\Rightarrow\lambda"] & C \arrow[dl, "g"] \\
& A &
\end{tikzcd}
=
\begin{tikzcd}
& \ell\downarrow C \arrow[d, dashed, "y"] \arrow[dl] \arrow[dr] & \\
B \arrow[dr, "f"'] & f\downarrow g \arrow[l] \arrow[r] \arrow[d, phantom, "\Rightarrow\mu'"] & C \arrow[dl, "g"] \\
& A &
\end{tikzcd}
=
\begin{tikzcd}
& \ell\downarrow C \arrow[dl] \arrow[dr] \arrow[d, phantom, "\Rightarrow\eta"] & \\
B \arrow[rr, "\ell"] \arrow[dr, "f"'] & {} \arrow[d, phantom, "\Rightarrow\lambda"] & C. \arrow[dl, "g"] \\
& A &
\end{tikzcd}
$$

Hence y and z define a fibered quasi equivalence and so in particular y is a fibered equivalence $\ell\downarrow C \simeq_{B\times C} f\downarrow g$.

Now conversely, suppose y is a fibered equivalence and let us argue that (ℓ, λ) is an absolute left lifting of f along g. Again we argue through a series of diagrams:

$$
\begin{tikzcd}
& X \arrow[dl] \arrow[dr] & \\
B \arrow[dr, "f"'] & \Rightarrow\alpha & C \arrow[dl, "g"] \\
& A &
\end{tikzcd}
=
\begin{tikzcd}
& X \arrow[d, dashed, "a"] \arrow[dl] \arrow[dr] & \\
B \arrow[dr, "f"'] & f\downarrow g \arrow[l] \arrow[r] \arrow[d, phantom, "\Rightarrow\mu'"] & C \arrow[dl, "g"] \\
& A &
\end{tikzcd}
=
\begin{tikzcd}
& X \arrow[d, "y^{-1}a"] \arrow[ddl, bend right] \arrow[ddr, bend left] & \\
& \ell\downarrow C \arrow[dl] \arrow[dr] \arrow[d, phantom, "\Rightarrow\eta"] & \\
B \arrow[rr, "\ell"] \arrow[dr, "f"'] & {} \arrow[d, phantom, "\Rightarrow\lambda"] & C \arrow[dl, "g"] \\
& A &
\end{tikzcd}
=
\begin{tikzcd}
& X \arrow[dl] \arrow[dr] \arrow[d, phantom, "\Rightarrow\alpha'"] & \\
B \arrow[rr, "\ell"] \arrow[dr, "f"'] & {} \arrow[d, phantom, "\Rightarrow\lambda"] & C \arrow[dl, "g"] \\
& A &
\end{tikzcd}
$$

We start with a square as above-left which we wish to uniquely factorize through λ. Applying the universal property of the comma type $f\downarrow g$ yields the center-left square. From center-left to center-right we use the inverse of the equivalence y, noting that the diagram remains commutative since y is fibered over $B \times C$, lastly, we whisker with the comma cone of $\ell\downarrow C$, giving $\alpha' :\equiv \eta\,\mathsf{pr}_0\, y^{-1} a$. Since each step preserves commutativity, this gives us a factorization of α through λ. Moreover, each step in this process was performed by an equivalence, first by the universal property of $f\downarrow g$, then by the fibered equivalence y, and finally by the universal property of $\ell\downarrow C$. Hence this factorization is unique and so (ℓ, λ) must be an absolute left lifting. $\square$

16.4 Cocartesian families

Definition 16.24 (Cocartesian arrow, [BW21, Definition 5.1.1])**.** Let $\pi : E \to B$ be an inner fibration of types. A **π-cocartesian arrow** ψ is an arrow $e \to_E e'$ such that all diagrams of the following form admit a unique lift

$$\begin{tikzcd}
\Delta^1 \arrow[d, hook, "t\mapsto\langle t,0\rangle"'] \arrow[dr, "\psi"] & \\
\Lambda_0^2 \arrow[r] \arrow[d, hook] & E \arrow[d, "\pi"] \\
\Delta^2 \arrow[r] \arrow[ur, dashed] & B.
\end{tikzcd}$$

Let P denote the straightening of π, then this may be typed as

$$\mathrm{isCocartArr}_\pi\,\psi :\equiv \prod_{\left(\sigma:\left\langle \Delta^2\to B\middle|^{s\equiv 0}_{\pi\circ\psi}\right\rangle\right)} \prod_{\left(h:\prod_{t:\Delta^1} P(\sigma\langle t,t\rangle)\right)} \mathsf{isContr}\left\langle \prod_{\langle t,s\rangle:\Delta^2} P(\sigma\langle t,s\rangle)\middle|^{\Lambda_0^2}_{[\psi,h]}\right\rangle$$

Alternatively, cocartesian arrows may be characterized by the following equivalence condition.

Remark 16.25 ([RV21, Definition 5.4.1])**.** An X-shaped arrow $\psi : X \to E^{\Delta^1}$ is an X-shaped π-cocartesian arrow if the induced dashed arrow in the following diagram is an equivalence, where j is the shape inclusion $\Lambda_0^2 \subseteq \Delta^2$ and i is the inclusion $t \mapsto \langle t, 0\rangle$ i.e. $s \equiv 0 \vdash s \leq t$,

$$\begin{tikzcd}
\psi/E \arrow[rr] \arrow[dr, dashed] \arrow[ddr, bend right, "\mathsf{pr}_0"'] & & E^{\Delta^2} \arrow[dr, "i\hat{\pitchfork}\pi"] \arrow[ddr, "E^{i_2}"' near end] & \\
& Q \arrow[rr] \arrow[d] \arrow[dr, phantom, "\lrcorner" very near start] & & i \rightrightarrows \pi \arrow[d] \\
& X \arrow[rr, "\psi"'] & & E^{\Delta^1}
\end{tikzcd}
\qquad \text{hence} \quad Q \simeq \sum_{x:X} \sum_{f:\left\langle \Lambda_0^2\to E\middle|^{s\equiv 0}_{\psi(x)}\right\rangle} \left\langle \Delta^2 \to B\middle|^{\Lambda_0^2}_{\pi\circ f}\right\rangle.$$

Definition 16.26 (Cocartesian fibrations and families, [BW21, Definition 5.2.1 and 5.2.2])**.** We say a fibration $\pi : E \to B$ with corresponding straightening $P : B \to \mathcal{U}$ **has cocartesial lifts** if we have a section

$$\mathrm{hasCocartLifts}\,\pi :\equiv \prod_{(b,b':B)} \prod_{(u:b\to_B b')} \prod_{(e:P(b))} \sum_{(e':P(b'))} \sum_{(\psi:e\to^P_u e')} \mathrm{isCocartArr}_\pi\,\psi,$$

and we define a cocartesian fibration to be an isoinner fibration with this property

$$\mathrm{isCocartFib}\,\pi :\equiv \mathrm{isIsoInner}\,\pi \times \mathrm{hasCocartLifts}\,\pi.$$

We introduce a series of glyphs for common 2-shapes for the purpose of the next proof.

Definition 16.27 (Glyphs for specific shapes)**.** We imagine the glyphs as miniatures for the shapes they represent with the s-axis pointing downward and centered dots reflecting an appropriate filler.

$$
\begin{aligned}
\ulcorner &:\equiv \{\langle t, s\rangle : \mathbb{2} \times \mathbb{2} \mid (t \equiv 0) \vee (s \equiv 0)\} && (\simeq \Lambda^2_0)\\
\urcorner &:\equiv \{\langle t, s\rangle : \mathbb{2} \times \mathbb{2} \mid (t \equiv 1) \vee (s \equiv 0)\} && (\simeq \Lambda^2_1)\\
\lrcorner &:\equiv \{\langle t, s\rangle : \mathbb{2} \times \mathbb{2} \mid (t \equiv 1) \vee (s \equiv 1)\} && (\simeq \Lambda^2_2)\\
\text{◺} &:\equiv \{\langle t, s\rangle : \mathbb{2} \times \mathbb{2} \mid (t \equiv 0) \vee (t \equiv s)\} && (\simeq \Lambda^2_0)\\
\sqcap &:\equiv \{\langle t, s\rangle : \mathbb{2} \times \mathbb{2} \mid (t \equiv 0) \vee (t \equiv 1) \vee (s \equiv 0)\} &&\\
\dot{\urcorner} &:\equiv \{\langle t, s\rangle : \mathbb{2} \times \mathbb{2} \mid s \leq t\} && (\simeq \Delta^2)\\
\dot{\llcorner} &:\equiv \{\langle t, s\rangle : \mathbb{2} \times \mathbb{2} \mid t \leq s\} && (\simeq \Delta^2)\\
\boxdot &:\equiv \{\langle t, s\rangle : \mathbb{2} \times \mathbb{2} \mid \top\} &&\\
\dot{\sqcap} &:\equiv \{\langle t, s\rangle : \mathbb{2} \times \mathbb{2} \mid (s \leq t) \vee (t \equiv 0)\} &&
\end{aligned}
$$

In particular we have the strict pushout squares

$$
\begin{tikzcd}
\urcorner \arrow[r, hook] \arrow[d, hook] & \dot{\urcorner} \arrow[d, hook] \\
\sqcap \arrow[r, hook] & \dot{\sqcap}
\end{tikzcd}
\quad \text{and} \quad
\begin{tikzcd}
\text{◺} \arrow[r, hook] \arrow[d, hook] & \dot{\llcorner} \arrow[d, hook] \\
\dot{\sqcap} \arrow[r, hook] & \boxdot .
\end{tikzcd}
$$

This is inspired by the notation of [RV21], although the meaning of our glyphs correspond only in part to theirs.

Theorem 16.28 ([RV21, Theorem 5.1.7(dual)])
Given an isoinner fibration $\pi : E \to B$ and an X-shaped arrow $\psi : X \to E^{\Delta^1}$, then the following are equivalent:

(i) The arrow ψ is π-cocartesian.

(ii) The following commutative triangle defines an absolute left lifting diagram

$$
\begin{tikzcd}
& E^{\Delta^1} \arrow[dr, "i_0 \hat{\pitchfork} \pi"] & \\
X \arrow[ur, "\psi"] \arrow[rr, "\pi\psi"'] \arrow[urr, phantom, "\|"] & & \pi \downarrow B
\end{tikzcd}
$$

(iii) There is an absolute left lifting diagram with $\mathsf{pr}_0\, \epsilon = \psi$ and $\mathsf{pr}_1\, \epsilon = \mathrm{id}_{\pi e}$

$$
\begin{tikzcd}
& E \arrow[dr, "\Delta_\pi"] & \\
X \arrow[ur, "e"] \arrow[rr, "\pi\psi"'] \arrow[urr, phantom, "\Uparrow\epsilon"] & & \pi \downarrow B
\end{tikzcd}
$$

Proof. We prove (i)⇒(ii)⇒(iii) leaving out (iii)⇒(i) due to time constraints, although we do believe that this last case may straightforwardly be adapted from [RV21] like the other cases. We note that this last case is also by far the most involved one.

(i)⇒(ii): By Theorem 16.23 it suffices to show the induced map y in the following diagram is an equivalence

$$
\begin{tikzcd}
& \psi \downarrow E^{\Delta^1} \arrow[dl, "\mathsf{pr}_0"'] \arrow[dr, "\mathsf{pr}_1"] \arrow[d, phantom, "="] & \\
X \arrow[rr, "\psi" description] \arrow[dr, "\pi\psi"'] & {} \arrow[d, phantom, "="] & E^{\Delta^1} \arrow[dl, "i_0 \widehat{\pitchfork} \pi"] \\
& B &
\end{tikzcd}
\quad = \quad
\begin{tikzcd}
& \psi \downarrow E^{\Delta^1} \arrow[dl, "\mathsf{pr}_0"'] \arrow[dr, "\mathsf{pr}_1"] \arrow[d, dashed, "y"] & \\
X & \pi\psi \downarrow (i_0 \,\widehat{\pitchfork}\, \pi) \arrow[l] \arrow[r] \arrow[d, phantom, "="] & E^{\Delta^1}. \arrow[dl, "i_0 \widehat{\pitchfork} \pi"] \\
& B \arrow[from=2-1, "\pi\psi"'] &
\end{tikzcd}
$$

In this case, the desired map displayed below-left is a pullback of the Leibniz cotensor of π with the inclusion $\iota : \sqcap \subseteq \boxdot$.

$$
\begin{tikzcd}
\psi \downarrow E^{\Delta^1} \arrow[rrr] \arrow[dr, dashed, "y"'] \arrow[ddr, bend right, "\mathsf{pr}_0"'] & & & E^{\boxdot} \arrow[d, "\iota \widehat{\pitchfork} \pi"] \arrow[dd, bend right, "\mathsf{pr}_0"'] \\
& \pi\psi \downarrow (i_0 \,\widehat{\pitchfork}\, \pi) \arrow[rr] \arrow[d] \arrow[drr, phantom, "\lrcorner" very near start] & & \iota \rightrightarrows \pi \arrow[d] \\
& X \arrow[rr, "\psi"'] & & E^{\Delta^1}
\end{tikzcd}
$$

The inclusion ι may be factorized as the composite of the inclusions $i : \sqcap \subseteq \sqcap$ and $j : \sqcap \subseteq \boxdot$. Hence we may apply Proposition 15.26 to get the following factorization of $\iota \,\widehat{\pitchfork}\, \pi$:

$$
\begin{tikzcd}
E^{\boxdot} \arrow[d, "j \widehat{\pitchfork} \pi"'] \arrow[dr, "\iota \widehat{\pitchfork} \pi"] & \\
j \rightrightarrows \pi \arrow[r, "\simeq"] \arrow[d] \arrow[dr, phantom, "\lrcorner" very near start] & \iota \rightrightarrows \pi \arrow[d] \\
E^{\sqcap} \arrow[r, "\simeq", "i \widehat{\pitchfork} \pi"'] & i \rightrightarrows \pi.
\end{tikzcd}
$$

Since i is a strict pushout of an inner horn inclusion and π is inner, the map $(j \rightrightarrows \pi) \to (\iota \rightrightarrows \pi)$ must be an equivalence. Moreover, since j is a strict pushout of the inclusion $\delta : \llcorner \subseteq \llcorner$, the leibniz cotensor $j \,\widehat{\pitchfork}\, \pi$ is a pullback of $\delta \,\widehat{\pitchfork}\, \pi$. By hypothesis, we know that $\delta \,\widehat{\pitchfork}\, \pi$ pulls back along ψ to an equivalence, thus $j \,\widehat{\pitchfork}\, \pi$ does as well. Now, by composability of equivalences this proves that $\iota \,\widehat{\pitchfork}\, \pi$ is an equivalence, hence y is as well.

(ii)⇒(iii): The unit of the adjunction $\mathsf{pr}_1 \dashv \Delta$ is an absolute left lifting by Corollary 16.15, and absolute left liftings compose by Lemma 16.19. Hence by observing that the following diagram is commutative we are done

$$\begin{tikzcd}
 & & E \arrow[dr, dashed, "\Delta"] \arrow[ddrr, bend left, "\Delta_\pi"] & & \\
 & E^{\Delta^1} \arrow[ur, dashed, "\mathrm{pr}_1"] \arrow[rr, equal] \arrow[rr, phantom, shift left=4ex, "\Uparrow"] \arrow[rr, phantom, shift right=4ex, "\|"] & & E^{\Delta^1} \arrow[dr, dashed, "i_0 \hat{\pitchfork} \pi"] & \\
X \arrow[ur, dashed, "\psi"] \arrow[uurr, bend left, "e"] \arrow[rrrr, "\pi\psi"'] & & & & \pi \downarrow B.
\end{tikzcd}$$

$\square$

References

[AK11] Peter Arndt and Krzysztof Kapulkin. "Homotopy-Theoretic Models of Type Theory". In: **Typed Lambda Calculi and Applications** (2011), pp. 45–60. ISSN: 1611-3349. DOI: 10.1007/978-3-642-21691-6_7. arXiv: 1208.5683 [math.LO].

[Bau13] Andrej Bauer. **Five Stages of Accepting Constructive Mathematics**. Institute for Advanced Study. Mar. 18, 2013. URL: https://www.ias.edu/video/members/1213/0318-AndrejBauer.

[BCP15] Marc Bezem, Thierry Coquand, and Erik Parmann. "Non-constructivity in Kan simplicial sets". In: **13th International Conference on Typed Lambda Calculi and Applications**. Vol. 38. LIPIcs. Leibniz Int. Proc. Inform. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2015, pp. 92–106.

[BM10] Clemens Berger and Ieke Moerdijk. "On an extension of the notion of Reedy category". In: **Mathematische Zeitschrift** 269.3-4 (Sept. 2010), pp. 977–1004. ISSN: 1432-1823. DOI: 10.1007/s00209-010-0770-x. arXiv: 0809.3341 [math.AT]. URL: http://dx.doi.org/10.1007/s00209-010-0770-x.

[BR13] Julia E. Bergner and Charles Rezk. "Reedy categories and the Θ-construction". In: **Math. Z.** 274.1-2 (2013), pp. 499–514. ISSN: 0025-5874. DOI: 10.1007/s00209-012-1082-0. arXiv: 1110.1066 [math.AT].

[Buc18] Ulrik Buchholtz. **Higher Structures in Homotopy Type Theory**. 2018. arXiv: 1807.02177 [math.LO].

[BW21] Ulrik Buchholtz and Jonathan Weinberger. **Synthetic fibered $(\infty, 1)$-category theory**. TU Darmstadt, 2021. arXiv: 2105.01724 [math.CT].

[DK80] W. G. Dwyer and D. M. Kan. "Function complexes in homotopical algebra". In: **Topology** 19.4 (1980), pp. 427–440. ISSN: 0040-9383. DOI: 10.1016/0040-9383(80)90025-7.

[DS95] William G. Dwyer and Jan Spaliński. "Homotopy theories and model categories". In: **Handbook of algebraic topology**. North-Holland, Amsterdam, 1995, pp. 73–126. DOI: 10.1016/B978-044481779-2/50003-1.

[Fri08] Greg Friedman. "Survey article: An elementary illustrated introduction to simplicial sets". In: **Rocky Mountain J. Math.** 42.2 (2008), pp. 353–423. ISSN: 0035-7596. DOI: 10.1216/RMJ-2012-42-2-353. arXiv: 0809.4221 [math.AT].

[GJ99] Paul G. Goerss and John F. Jardine. **Simplicial homotopy theory**. Vol. 174. Progress in Mathematics. Birkhäuser Verlag, Basel, 1999, pp. xvi+510. ISBN: 3-7643-6064-X. DOI: 10.1007/978-3-0348-8707-6.

[GSS19] Nicola Gambino, Christian Sattler, and Karol Szumiło. **The constructive Kan–Quillen model structure: two new proofs**. 2019. arXiv: 1907.05394 [math.AT].

[GZ67] P. Gabriel and M. Zisman. **Calculus of fractions and homotopy theory**. Ergebnisse der Mathematik und ihrer Grenzgebiete, Band 35. Springer-Verlag New York, Inc., New York, 1967, pp. x+168.

[HKRS17] Kathryn Hess et al. "A necessary and sufficient condition for induced model structures". In: **Journal of Topology** 10.2 (Apr. 2017), pp. 324–369. ISSN: 1753-8424. DOI: 10.1112/topo.12011. arXiv: 1509.08154 [math.AT].

[HS98] Martin Hofmann and Thomas Streicher. "The groupoid interpretation of type theory". In: **Twenty-five years of constructive type theory (Venice, 1995)**. Vol. 36. Oxford Logic Guides. New York: Oxford Univ. Press, 1998, pp. 83–111.

[Joy08] André Joyal. **The theory of quasi-categories and its applications**. Centre de Recerca Matemática, 2008. URL: `https://ncatlab.org/nlab/files/JoyalTheoryOfQuasiCategories.pdf`.

[Joy21] André Joyal. **Model categories**. Dec. 2021. URL: `https://ncatlab.org/joyalscatlab/show/Model+categories`.

[KL18] Krzysztof Kapulkin and Peter LeFanu Lumsdaine. **The Simplicial Model of Univalent Foundations (after Voevodsky)**. 2018. arXiv: `1211.2851 [math.LO]`.

[Kra21] Nicolai Kraus. **Internal ∞-Categorical Models of Dependent Type Theory: Towards 2LTT Eating HoTT**. 2021. arXiv: `2009.01883 [cs.LO]`.

[Law06] F. William Lawvere. "Adjointness in foundations". In: **Repr. Theory Appl. Categ.** 16 (2006). Reprinted from Dialectica **23** (1969), pp. 1–16.

[Lur09] Jacob Lurie. **Higher topos theory**. Vol. 170. Annals of Mathematics Studies. Princeton University Press, Princeton, NJ, 2009, pp. xviii+925. ISBN: 978-0-691-14049-0; 0-691-14049-9. DOI: `10.1515/9781400830558`. URL: `https://www.math.ias.edu/~lurie/papers/HTT.pdf`.

[nLaba] nLab authors. **categorical model of dependent types**. Oct. 2021. URL: `http://ncatlab.org/nlab/show/categorical%20model%20of%20dependent%20types`.

[nLabb] nLab authors. **weak factorization system**. Oct. 2021. URL: `http://ncatlab.org/nlab/show/weak%20factorization%20system`.

[nLabc] nLab authors. **locally presentable category**. Oct. 2021. URL: `http://ncatlab.org/nlab/show/locally%20presentable%20category`.

[nLabd] nLab authors. **canonical model structure on Cat**. Nov. 2021. URL: `http://ncatlab.org/nlab/show/canonical%20model%20structure%20on%20Cat`.

[Qui67] Daniel G. Quillen. **Homotopical algebra**. Lecture Notes in Mathematics, No. 43. Springer-Verlag, Berlin-New York, 1967, iv+156 pp. (not consecutively paged).

[Ras18] Nima Rasekh. **Introduction to Complete Segal Spaces**. 2018. arXiv: `1805.03131 [math.CT]`.

[RCS18] Emily Riehl, Evan Cavallo, and Christian Sattler. **On the directed univalence axiom**. Talk at the AMS Special Session on Homotopy Type Theory, Joint Mathematics Meetings. AMS, 2018. URL: `https://math.jhu.edu/~eriehl/JMM2018-directed-univalence.pdf`.

[Rez01] Charles Rezk. "A model for the homotopy theory of homotopy theory". In: **Trans. Amer. Math. Soc.** 353.3 (2001), pp. 973–1007. ISSN: 0002-9947. DOI: `10.1090/S0002-9947-00-02653-2`. arXiv: `9811037 [math.AT]`.

[Rie09] Emily Riehl. **A concise definition of a model category**. Sept. 3, 2009. URL: `https://math.jhu.edu/~eriehl/modelcat.pdf`.

[Rie20] Emily Riehl. **Homotopical categories: from model categories to (∞,1)-categories**. 2020. arXiv: `1904.00886 [math.AT]`. URL: `https://math.jhu.edu/~eriehl/homotopical.pdf`.

[RS17] Emily Riehl and Michael Shulman. "A type theory for synthetic ∞-categories". In: **High. Struct.** 1.1 (2017), pp. 147–224. DOI: 10.1007/s42001-017-0005-6. arXiv: 1705.07442 [math.CT]. URL: https://math.jhu.edu/~eriehl/synthetic.pdf.

[RV14] Emily Riehl and Dominic Verity. "The theory and practice of Reedy categories". In: **Theory Appl. Categ.** 29 (2014), pp. 256–301. arXiv: 1304.6871 [math.CT].

[RV16] Emily Riehl and Dominic Verity. "Homotopy coherent adjunctions and the formal theory of monads". In: **Adv. Math.** 286 (2016), pp. 802–888. ISSN: 0001-8708. DOI: 10.1016/j.aim.2015.09.011. arXiv: 1310.8279 [math.CT].

[RV21] Emily Riehl and Dominic Verity. **Elements of ∞-Category Theory**. Cambridge University Press, 2021. URL: https://math.jhu.edu/~eriehl/elements.pdf.

[Shu19] Michael Shulman. **All $(\infty, 1)$-toposes have strict univalent universes**. 2019. arXiv: 1904.07004 [math.AT].

[Soj21] Kristina Sojakova. **Syllepsis in Homotopy Type Theory**. 2021. arXiv: 2107.14283 [cs.LO].

[Str14] Thomas Streicher. "A model of type theory in simplicial sets: a brief introduction to Voevodsky's homotopy type theory". In: **J. Appl. Log.** 12.1 (2014), pp. 45–49. ISSN: 1570-8683. DOI: 10.1016/j.jal.2013.04.001.

[SU06] Morten Heine B. Sørensen and Paweł Urzyczyn. **Lectures on the Curry-Howard Isomorphism**. Vol. 149. Studies in Logic and the Foundations of Mathematics. Elsevier B. V., July 4, 2006. ISBN: 978-0-444-52077-7. URL: https://disi.unitn.it/~bernardi/RSISE11/Papers/curry-howard.pdf.

[UF13] The Univalent Foundations Program. **Homotopy type theory—univalent foundations of mathematics**. The Univalent Foundations Program, Princeton, NJ; Institute for Advanced Study (IAS), Princeton, NJ, 2013, pp. xiv+589. URL: https://homotopytypetheory.org/book.

[Voe14] Vladimir Voevodsky. **The Origins and Motivation of Univalent Foundations**. 2014. URL: https://www.ias.edu/ideas/2014/voevodsky-origins.

[Wer97] Benjamin Werner. "Sets in types, types in sets". In: **Theoretical aspects of computer software (Sendai, 1997)**. Vol. 1281. Lecture Notes in Comput. Sci. Springer, Berlin, 1997, pp. 530–546. DOI: 10.1007/BFb0014566.

[WL20] Matthew Z. Weaver and Daniel R. Licata. "A constructive model of directed univalence in bicubical sets". In: **Proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2020)**. ACM, New York, July 2020, pp. 915–928. DOI: 10.1145/3373718.3394794. URL: https://dlicata.wescreates.wesleyan.edu/pubs/wl20bicubical/wl20bicubical.pdf.

Fredrik Bakke

Segal Spaces in Homotopy Type Theory

NTNU

Norwegian University of
Science and Technology