Université Paris Cité

École de Sciences Mathématiques Paris Centre (ED 386)
Institut de Recherche Fondamentale en Informatique (IRIF)

Thèse de doctorat en Informatique
Dirigée par Hugo HERBELIN

Cubical Models Are Cofreely Parametric

par Hugo MOENECLAEY

Présentée et soutenue publiquement le 21 octobre 2022
devant un jury composé de :

Steve AWODEY* (Prof., Carnegie Mellon University)	Rapporteur
Thorsten ALTENKIRCH (Prof., University of Nottingham)	Rapporteur
Peter DYBJER (Prof., Chalmers University of Technology)	Examinateur
Eric FINSTER (Lecturer, University of Birmingham)	Examinateur
Patricia JOHANN (Professor, Appalachian State University)	Examinatrice
Ambrus KAPOSI (Assoc. Prof., Eötvös Loránd Universit)	Examinateur
Muriel LIVERNET (Prof., Université Paris Cité)	Examinatrice
Hugo HERBELIN (DR, Université Paris Cité)	Directeur

*en visioconférence

Les modèles cubiques sont colibrement paramétriques

Hugo Moeneclaey

Abstract. A parametric model of type theory is defined as a model where any type comes with a relation and any term respects these. Intuitively, this means that terms treat their inputs uniformly.

In recent years many cubical models of type theory have been proposed, often built to support some form of parametricity. In this thesis, we explain this phenomena by defending that cubical models of type theory are cofreely parametric. To do this, we define notions of parametricity and their associated parametric models, then we prove that cofreely parametric models exist, and finally we give examples of cubical models which are indeed cofreely parametric.

In Chapter 1, we define the standard parametricity in details for categories and clans, with homotopically-flavored examples of parametric models. Then we give an informal survey of variants of parametricity, giving us ample potential applications for the next chapters. An important variant is internal parametricity where any type comes with a reflexive relation.

In Chapter 2, we axiomatize the situation by going back to the historical approach to parametricity, namely that it is inductively proven for the initial model. So an extension by section of a theory is defined as an extension by inductively defined unary operations. This is made precise using signatures for quotient inductive-inductive types. The extensions of the theory of categories, clans and categories with families by the standard parametricity are all key examples of extensions by section. We prove that the forgetful functors coming from such extensions have right adjoints, so that cofreely parametric models exist. We also explain how to extend the standard parametricity to arrow types and universes.

In Chapter 3 we give an alternative axiomatization of parametricity, that manages to give a very compact description for cofreely parametric models when applicable. We work with a symmetric monoidal closed category $\mathcal{V}$ of models of type theory. We define a notion of parametricity as a monoid in $\mathcal{V}$, and a parametric model as a module. Then we build cofreely (and freely) parametric models as coinduced (and induced) modules. We prove that strict variants of both the category of left exact categories and the category of clans are symmetric monoidal closed. Then we prove that both the lex categories of n-truncated cubical objects and the clans of Reedy fibrant cubical objects are cofreely parametric models for suitable notions of parametricity.

Keywords. Dependent type theory, Parametricity, Cubical models for type theory, Cofree objects.

RÉSUMÉ. Un modèle paramétrique de la théorie des types est défini comme un modèle où chaque type est muni d'une relation, et les termes respectent ces relations. Intuitivement, cela veut dire que les termes traitent leurs entrées uniformément.

Ces dernières années, de nombreux modèles cubiques de la théorie des types ont été proposés, souvent conçus pour valider une variante de paramétricité. Dans cette thèse, on explique ce phénomène en prouvant que les modèles cubiques sont colibrement paramétriques. Pour cela on définit les notions de paramétricité et leurs modèles paramétriques associés. On prouve ensuite que les modèles colibrement paramétriques existent, puis on donne des exemples de modèles cubiques qui sont colibrement paramétriques.

Dans le chapitre 1, on définit la paramétricité standard pour les catégories et les clans. On donne ensuite des exemples de modèles paramétriques inspirés par la théorie de l'homotopie. On présente informellement les variantes de paramétricité qui seront formalisées dans les chapitres suivants. La paramétricité interne est l'une de ces variantes particulièrement notable, où chaque type est muni d'une relation réflexive.

Dans le chapitre 2, on donne une axiomatisation de cette situation inspirée par l'approche originelle de la paramétricité, c'est-à-dire inspirée du fait que l'on peut prouver le modèle initial paramétrique par induction. Plus précisément, on définit une extension par section d'une théorie comme une extension par des opérations unaires définies inductivement. On utilise pour cela la théorie des signatures pour les types inductifs-inductifs quotients. Les extensions de la théorie des catégories, des clans ou des catégories avec famille par la paramétricité standard sont des exemples importants d'extensions par section. On prouve ensuite que les foncteurs d'oubli provenant de telles extensions ont des adjoints à droite, et donc que les modèles colibrement paramétriques existent. On explique comment étendre la paramétricité standard aux types de fonctions ainsi qu'aux univers.

Dans le chapitre 3, on donne une axiomatisation alternative de la paramétricité, qui permet une description très compacte des modèles colibrement paramétriques. On postule d'abord une catégorie symétrique monoïdale fermée $\mathcal{V}$ de modèles de la théorie des types. On définit alors une notion de paramétricité comme un monoïde dans $\mathcal{V}$, et un modèle paramétrique comme un module. On peut donc définir les modèles colibrement (et librement) paramétriques comme des modules coinduits (et induits). On prouve ensuite que des variantes strictes de la catégorie des catégories exactes à gauche et de la catégorie des clans sont symétriques monoïdales fermées. On prouve finalement que les catégories exactes à gauche d'objets cubiques tronqués, ainsi que les clans d'objets cubiques fibrant au sens de Reedy, sont colibrement paramétriques pour des notions de paramétricité appropriées.

Mots-clés. Théorie des types dépendants, Paramétricité, Modèles cubiques de la théorie des types, Objets colibres.

Contents

Introduction

This thesis is concerned with cubical models for type theories, and their links with parametricity.

0.1. Type theories and parametricity

Type theories form a family of foundational systems for mathematics based on Martin-Löf type theory [**ML75, MLS84**]. In such foundations both sets and propositions are modeled by the so-called types, and both elements of sets and proofs of propositions are modeled by terms in these types. In this thesis, we adopt a semantical point of view on type theories, mostly following [**Dyb95**] and [**Joy17**] (see [**Hof97**] for a gentle introduction). This means that in order to study a type theory (i.e. a family of rules building types and terms), we study its models (i.e. notions of types and terms obeying these rules).

Reynolds introduced parametricity for system F in [**Rey83**]. There he proved inductively on types and terms in system F that:

(1) Any type comes with a relation on its terms.
(2) Any term respects these relations, meaning that substituting related variables in a term gives related terms.

In this thesis, relations are always understood as binary unless indicated otherwise. Parametricity is useful to prove that terms in system F are well-behaved in various ways [**Wad89**]. We say that they are parametric, as they treat their inputs uniformly. By contrapositive, we can prove that some functions are not definable in system F, by proving that they are not parametric.

Parametricity has been extended to various type theories [**Tak01, BJP10, BL11, KL12b**], meaning that any type in these theories comes with a relation, and any term preserves these. A pleasant feature of these extensions is that relations can be asserted inside type theory using dependent types, whereas the relations for system F had to be defined in another theory (e.g. set-theory for Reynolds). This is studied in depth in [**BL11**], which gives a general method taking as input a pure type system P, and giving as output another pure type system P^2 suitable to express the parametricity of P.

In this thesis we will define and study parametric models of type theory. The relations assumed in such a model should respect the structure of the model, for example the relation over a product $A \times B$ should be the product of the relations over A and over B.

0.2. Cubes and parametricity

A cubical structure on a type consists of:

- For any two elements a type of so-called paths between them.
- Given four paths drawing a square, a type of fillers for this square.

- And so on, defining fillers for cubes of any dimensions.

When trying to build parametric models of type theories, cubical structures often arise. For example:

- In [**BCM15**] a model of type theory obeying (a variant of) parametricity is built using (a variant of) cubical sets.
- In [**JS17**] some models for n-iterated versions of Reynolds parametricity (called n-dimensional) are built using n-truncated cubes with reflexivities.
- In [**CH20**] a variant of cubical type theory is introduced, that is a type theory where a cubical structure on types is internalised. It supports a form of parametricity. This is strikingly similar to the cubical type theory in [**CCHM15**] supporting univalence.

We want to explain this phenomenon by defending the following thesis:

Thesis 1. *Cubical models for type theory are cofreely parametric.*

More precisely, we claim that for many notions of model of type theory, and many variants of cubical structure, there exists a notion of parametricity such that we have an adjunction:

$$\begin{tikzcd}
\{\text{Models of type theory}\} \arrow[rr, bend right, "\mathcal{C} \mapsto \{\text{cubical types in } \mathcal{C}\}"'] \arrow[rr, phantom, "\perp"] & & \{\text{Parametric models}\} \arrow[ll, bend right, "\text{forgetful functor}"']
\end{tikzcd}$$

In a parametric model, any type comes with a relation. But this relation is itself a type, so it comes with a relation over it, and so on. The main insight leading to the previous adjunction is that by iterating this process we get a cubical type.

We give a simple example as an illustration:

Definition 0.2.1. A parametric category is a category $\mathcal{C}$ equipped with:

- An endofunctor:
$$\Gamma \mapsto \Gamma_* \tag{0.2.1}$$
of $\mathcal{C}$.
- For any Γ in $\mathcal{C}$ two morphisms:
$$d^0_\Gamma, d^1_\Gamma \quad : \quad \Gamma_* \to \Gamma \tag{0.2.2}$$
natural in Γ.

So a parametric category is a category where any object Γ comes with a relation internal to $\mathcal{C}$ as follows:

$$(d^0_\Gamma, d^1_\Gamma) \quad : \quad \Gamma_* \to \Gamma \times \Gamma \tag{0.2.3}$$

and any morphism respects these relations.

Then the forgetful functor from parametric categories to categories has a right adjoint, sending $\mathcal{C}$ to the category of semi-cubical (i.e. cubical with face maps only) objects in $\mathcal{C}$. In this case the endofunctor sends a semi-cubical object Γ to the semi-cubical object Γ_* of paths in Γ, with d^0_Γ (resp. d^1_Γ) sending a path to its source (resp. target).

We also want to argue that cofreely parametric models tend to exist. This will be supported by proving the existence of many right adjoints to forgetful functors, building not only cubical structures, but also structures based on similar (or not so similar) shapes.

Remark 0.2.2. All functors forgetting parametricity have left adjoints. These allow to built freely parametric models, which are very different from cofreely parametric ones.

- The freely parametric model generated by $\mathcal{C}$ simply assume $\mathcal{C}$ parametric. This brutal process leads to an incoherent model if $\mathcal{C}$ has a term contradicting parametricity.
- The cofreely parametric model generated by $\mathcal{C}$ is the largest parametric fragment of $\mathcal{C}$. When $\mathcal{C}$ has a term contradicting parametricity, it will simply not occur in the cofreely parametric model, without collapsing everything.

0.3. Variants of parametricity

We give an overview for the various notions of parametricity, and their role in this thesis.

- **Arity.** We assumed that any type comes with a binary relation, but we could have assumed a predicate (yielding augmented simplices rather than cubes) or more generally an n-ary relation. This does not cause any issue, and we will use unary parametricity when it leads to simpler notations.

- **External / internal.** The standard parametricity is called external, because it cannot be used inside type theory. Indeed we cannot prove that variables are related to themselves, so that we cannot prove that non-closed terms are related to themselves. Internal parametricity attempts to correct this by introducing reflexivities, that is witnesses that any variable in a type is related to itself. In this case the right adjoint yield cubes, that are semi-cubes with reflexivites.

  Remark 0.3.1. Reflexivities together with arrow types or universes do not fit in our framework, because we do not know how to define reflexivity for:

  $$A \to B \tag{0.3.1}$$

  inductively from reflexivities for A and B, or to define reflexivity for a type variable in the universe. See Remark 2.7.8 for more details.

- **Iterated / truncated.** The standard parametricity is iterated, in the sense that it can be applied as many times as we want, building a cubical structure. It is also possible to define truncated variants of parametricity, which can only be applied a fixed number of times. As an example, graphs (i.e. 1-truncated cubical structure) are obtained from a parametricity that can be applied only once. Both [**AGJ14**] and [**GFO16**] deal explicitly with truncated notions of parametricity.

- **Relative / univalent.** We call the standard parametricity relative because it asserts the existence of relations. The variants asserting equivalences (meaning here relations with transports) are called univalent. They should lead to right adjoints building Kan cubical structures. Such a univalent variant of parametricity has been considered in [**AK17**], and recently extended in collaboration with Michael Shulman under the name of Higher Observational Type Theory (unpublished at the moment). In [**CH20**], relative parametricity and univalence are introduced in a parallel fashion, emphasizing the similarities between the two. These approaches do not fit in our framework as they use reflexivities and arrow types. Nevertheless they are key motivators in our study of parametricity. See [**TTS21**] for an alternative approach assuming a univalent universe to begin with, which seems to fit in our framework.

Our goal is to prove that cofreely parametric models exist for such variants. Moreover, we want to give explicit descriptions for the right adjoints when possible. As models of type theory, we will mainly consider category with families [**Dyb95**] (see [**CCD21**] for an up-to-date introduction) and clans [**Joy17**]. We do not expect any issue when extending this thesis to other essentially algebraic notions of models of type theory, at least in the absence of arrow types and universes.

0.4. Notions of parametricity as extensions by section

Assume given a theory T of models of type theory (for example T is the theory of categories with families). Here by theory we mean a signature for quotient inductive-inductive types [**KKA19**] (see [**AK16**] for an early account focusing on models of type theory). This could be adapted to an essentially algebraic theory [**AHR99**] or a generalized algebraic theory [**Car86**]. The theory T' of parametric models is an extension of T of a very specific form. We axiomatize this form under the name of *extension by section*.

Definition 0.4.1. An extension by section of a theory T is an extension by:

- Unary operations with equations defining them inductively.
- Inductively provable unary equations.

So the theory of parametric models is an extension by section of the theory of models for type theory. See Section 2.2 for more details.

Remark 0.4.2. We introduced extensions by section and in [**Moe21**] under the somewhat bland name of *interpretations*. In Chapter 2 of this thesis we give essentially the same results as in [**Moe21**], but with new definitions and proofs. The examples largely overlap between these two accounts.

The main result from Chapter 2 is as follows:

THEOREM 0.4.3. *Assume given an extension by section T' of T. Then the forgetful functor:*

$$U \quad : \quad \mathrm{Alg}_{T'} \to \mathrm{Alg}_T \tag{0.4.1}$$

has a right adjoint.

In [**Moe21**] the existence of this right adjoint was proved using the theory of locally presentable categories [**AR94**]. The streamlined proof given in this thesis uses a direct definition of cofree objects as limits.

Our main examples of such an extension by section is standard parametricity for categories, clans and categories with families with or without arrow types and a universe.

0.5. Notions of parametricity as monoidal models

In Chapter 3 we give an alternative approach, which yields more compact descriptions for cofreely and freely parametric models. This will allow us to prove that many cubical models are indeed cofreely parametric.

For any theory T, the theory of T-algebras with an endomorphism is an extension by section of T. Inspired by this example, we will consider extensions adding a module structure to an object.

We assume a symmetric monoidal closed category $\mathcal{V}$ of models of type theory, and for $\mathcal{C}$ and $\mathcal{D}$ in $\mathcal{V}$ we denote by:

$$\mathcal{C} \multimap \mathcal{D} \tag{0.5.1}$$

the exponential of $\mathcal{D}$ by $\mathcal{C}$. Then a notion of parametricity for $\mathcal{V}$ is defined as a monoid $\mathcal{M}$ in $\mathcal{V}$. An $\mathcal{M}$-parametric model is defined as an $\mathcal{M}$-module in $\mathcal{V}$. We can construct freely and cofreely parametric objects simply as induced and coinduced modules, so that we have a string of adjoint functors:

$$\begin{tikzcd} \mathcal{V} \arrow[r, bend left, "\mathcal{C} \mapsto \mathcal{M}\otimes\mathcal{C}"] \arrow[r, bend right, "\mathcal{C} \mapsto \mathcal{M}\multimap\mathcal{C}"'] & \{\mathcal{M}\text{-modules}\} \arrow[l] \end{tikzcd}$$

where the $\mathcal{M}$-module structure on $\mathcal{M} \otimes \mathcal{C}$ (resp. $\mathcal{M} \multimap \mathcal{C}$) is induced by the left (resp. right) $\mathcal{M}$-module structure on $\mathcal{M}$ induced by multiplication.

Example 0.5.1. We consider $\mathcal{V}$ the cartesian closed category of categories. Any category of cubes $\square$ in [**BM17**] is monoidal. It can have diagonals, reflexivities, symmetries, connexions or reversals. We get that $\mathcal{C}^{\square}$ is a cofreely $\square$-parametric category for any such category $\square$.

As an example, the standard parametricity is obtained by considering as $\square$ the monoidal category generated by an object $\mathbb{I}$ with two morphisms:

$$d^0, d^1 \quad : \quad \mathbb{I} \longrightarrow 1 \tag{0.5.2}$$

This category is equivalent to (the opposite of) the category of semi-cubes.

Example 0.5.2. We will consider $\mathcal{V}$ to be the category of (a strict variant of) left exact categories, abbreviated by strict lex categories. In this case a notion of parametricity (i.e. a monoid in $\mathcal{V}$) is a strict lex category with a monoidal product commuting with finite limits in both variables.

As an example we will prove that the lex category generated by:

$$s, t \quad : \quad E \longrightarrow V \tag{0.5.3}$$

is in fact a notion of parametricity where:

- The object V is the unit for the monoidal product.
- The product $E \otimes E$ is isomorphic to the pullback of four copies of E drawing a square.

Then lex categories of graphs (that is of 1-truncated semi-cubical objects) are cofreely parametric for this notion of parametricity. This will be extended to n-truncated semi-cubes and cubes with reflexivities.

Example 0.5.3. We will consider $\mathcal{V}$ the category of (a strict variant of) clans. A monoid in $\mathcal{V}$ will be a clan with a monoidal product:

- Commuting with limits in both variables.
- Such that given two fibrations:

$$i \twoheadrightarrow i' \tag{0.5.4}$$

$$j \twoheadrightarrow j' \tag{0.5.5}$$

  we have an induced fibration:

$$i \otimes j \twoheadrightarrow (i' \otimes j) \underset{i' \otimes j'}{\times} (i \otimes j') \tag{0.5.6}$$

As an example, we will prove that the clans of Reedy fibrant semi-cubes are cofreely parametric, using as notion of parametricity the monoidal clan generated by an object $\mathbb{I}$ with a fibration:

$$\mathbb{I} \twoheadrightarrow 1 \times 1 \tag{0.5.7}$$

The same holds for cubes with reflexivities.

0.6. Plan

This thesis is organised in three chapters.

Chapter 1 is an introduction to parametricity and models of type theory:

- In Sections 1.1 to 1.3 the standard parametricity, where any type comes with a relation, is introduced for categories and clans. This presents the categorical point of view on parametricity used in this thesis.
  - An introduction to clans is given in Section 1.2.
  - Some homotopically-flavored examples of parametric models are given in Section 1.1 for categories and Section 1.3 for clans, emphasizing the relevance of parametricity to homotopically-minded readers.
- An introduction to category with families is given in Section 1.4. Parametric categories with families are not introduced at this point.
- The main variants of parametricity are surveyed in Section 1.5, with reference to the literature. This section contains some vocabulary used in the rest of the thesis.

Chapter 2 is about extensions by section:

- Section 2.1 introduces categorical extensions by section, which are a specific kind of forgetful functors, and prove that they have right adjoints.
- Signatures for quotient inductive-inductive types are surveyed in Section 2.2, with pointers to [**KKA19**] and to [**KK20**] for details. These signatures are then used to define extensions by section.
- Section 2.3 contains the main result from this chapter, namely that the forgetful functor associated to an extension by section has a right adjoint. This result was already proved in [**Moe21**], but we give a new proof here using categorical extensions by section.

- In Sections 2.4 to 2.7 we prove that the standard notions of parametricity for categories, clans and categories with families (with or without arrow types and a universe) are indeed extensions by section. The most technical parts of the proofs for categories with families can be found in Appendix A.
- In Section 2.8 we sketch two conjectural examples of extensions by section, both concerned with restricted forms of univalence. First we explain why we expect *setoid type theory* [**ABKT19**] to model a form of 1-truncated univalent parametricity, excluding arrow types and universe for sets. Then we consider *univalent parametricity* from [**TTS21**].

Chapter 3 is about parametric models defined as modules:

- Sections 3.1 to 3.3 axiomatize notions of parametricity as monoids, and proves that cofreely parametric models exist in this context.
  - Section 3.1 serves as a motivation. It explains how parametric categories as in Section 1.1 can be seen as modules over a monoidal category. This will be a running example in the two following sections.
  - Notions of parametricity and parametric models are defined as monoids and modules in Section 3.2. The definitions of monoids and modules in an arbitrary monoidal category are recalled.
  - In Section 3.3 we prove that functors forgetting a module structure have both left and right adjoints. The given proof uses the interpretation of the multiplicative fragment of linear logic in symmetric monoidal closed categories, although a direct proof would be possible.
- This axiomatization is used for lex categories in Sections 3.4 to 3.6.
  - In Section 3.4, we prove that lex categories form a symmetric monoidal closed category. For this to hold we need to consider a particularly strict version of left categories: functors commute with limits on the nose, and the canonical isomorphisms asserting the commutation of limits are assumed to be identities. The monoids in this category of strict lex categories (also called the monoidal strict lex categories) are made explicit in Section 3.5.
  - Some truncated notions of parametricity for lex categories are defined in Section 3.6. More precisely, lex categories of n-truncated semi-cubical (or cubical with reflexivities) objects are proven to be cofreely parametric.
- This axiomatization is used for clans in Sections 3.7 to 3.9.
  - Sections 3.7 and 3.8 contain an adaptation of Sections 3.4 and 3.5 from lex categories to clans. So all definitions and proofs have to be extended to deal with fibrations, giving a symmetric monoidal closed category of strict clans.
  - In Section 3.9, clans of Reedy fibrant semi-cubical (or cubical with reflexivities) objects are proven to be cofreely parametric.

Introduction (français)

On s'intéresse dans cette thèse aux modèles cubiques des théories des types, et à leurs liens avec la paramétricité.

0.1. Théories des types et paramétricité

Les théories des types forment une famille de systèmes formels qui peuvent servir de fondation pour les mathématiques, basés sur la théorie des types de Martin-Löf [**ML75, MLS84**]. Dans de tels systèmes, les ensembles et les propositions sont modélisés par des types, et les éléments des ensembles ainsi que les preuves des propositions sont modélisés par des termes dans ces types. Dans cette thèse, on adopte un point de vue sémantique sur les théories des types, en suivant principalement [**Dyb95**] et [**Joy17**] (voir [**Hof97**] pour une introduction). Cela signifie que pour étudier une théorie des types (c'est-à-dire une famille de règles permettant de construire des types et des termes), on étudie ses modèles (c'est-à-dire les notions de types et de termes obéissant à ces règles).

La paramétricité a été introduite pour le système F par Reynolds dans [**Rey83**]. On y trouve une preuve par induction sur les types et les termes du système F que :

(1) Tout type est équipé d'une relation sur ses termes.
(2) Tout terme respecte ces relations, ce qui signifie que la substitution de variables reliées dans un terme donne des termes reliés.

Dans cette thèse, les relations sont toujours comprises comme binaires, sauf indication contraire. La paramétricité permet de prouver que les termes du système F se comportent bien [**Wad89**]. On dit que ces termes sont paramétriques, car ils traitent leurs entrées de manière uniforme. Par contraposition, on peut prouver que certaines fonctions ne sont pas définissables dans le système F en prouvant qu'elles ne sont pas paramétriques.

La paramétricité a été étendue à diverses théories des types [**Tak01, BJP10, BL11, KL12b**], ce qui signifie que tout type dans ces théories est équipé d'une relation, et que tout terme les préserve. Ces extensions sont particulièrement plaisantes car les relations peuvent être définies à l'intérieur de la théorie des types, en utilisant des types dépendants, alors que les relations pour le système F devaient être définies dans un autre système (par exemple la théorie des ensembles pour Reynolds). L'article [**BL11**] présente une méthode générale prenant en entrée un système de type pur P, et donnant en sortie un autre système de type pur P^2 approprié pour exprimer la paramétricité de P.

Dans cette thèse, on va définir et étudier des modèles paramétriques de la théorie des types. Les relations supposées dans un tel modèle doivent respecter la structure du modèle, par exemple la relation sur un produit $A \times B$ doit être le produit des relations sur A et sur B.

0.2. Cubes et paramétricité

Une structure cubique sur un type consiste en la donnée de :

- Pour deux termes quelconques, un type des chemins entre eux.
- Étant donnés quatre chemins dessinant un carré, un type des surfaces remplissant ce carré.
- Et ainsi de suite, en définissant les types des remplissages pour les cubes de dimension quelconque.

Lorsque l'on essaie de construire des modèles paramétriques de la théorie des types, on rencontre souvent des structures cubiques. Par exemple :

- Dans [**BCM15**], un modèle de la théorie des types satisfaisant une variante de paramétricité est construit en utilisant une variante des ensembles cubiques.
- Dans [**JS17**], des modèles pour les versions itérées n fois de la paramétricité de Reynolds sont construits, en utilisant des catégories cubiques n-tronquées.
- Dans [**CH20**], une variante de la théorie des types cubique est introduite, c'est-à-dire une théorie des types où une structure cubique sur les types est internalisée. Elle satisfait une forme de paramétricité. Elle est remarquablement similaire à la théorie des types cubique de [**CCHM15**], qui satisfait l'univalence.

On explique ce phénomène en défendant la thèse suivante :

THÈSE 1. *Les modèles cubiques de la théorie des types sont colibrement paramétriques.*

Plus précisément, on affirme que pour de nombreuses notions de modèle de la théorie des types, et de nombreuses variantes de structure cubique, il existe une notion de paramétricité telle que l'on a une adjonction :

$$\begin{tikzcd}
\{\text{Modèles de la théorie des types}\} \arrow[rr, bend right, "\mathcal{C} \mapsto \{\text{types cubiques dans } \mathcal{C}\}"'] \arrow[rr, phantom, "\perp"] & & \{\text{Modèles paramétriques}\} \arrow[ll, bend right, "\text{foncteur d'oubli}"']
\end{tikzcd}$$

Dans un modèle paramétrique, tout type est équipé d'une relation. Mais cette relation est elle-même un type, elle est donc à son tour équipée d'une relation, et ainsi de suite. L'idée principale derrière l'adjonction précédente est que l'on obtient un type cubique en itérant ce processus.

On donne un exemple simple, à titre d'illustration :

Définition 0.2.1. Une catégorie paramétrique est une catégorie $\mathcal{C}$ équipée :

- D'un endofoncteur :
$$\Gamma \mapsto \Gamma_* \tag{0.2.1}$$
de $\mathcal{C}$.
- Pour tout Γ dans $\mathcal{C}$, de deux morphismes :
$$d^0_\Gamma, d^1_\Gamma \quad : \quad \Gamma_* \to \Gamma \tag{0.2.2}$$

naturels en Γ.

Une catégorie paramétrique est donc une catégorie où tout objet Γ est équipé d'une relation interne à $\mathcal{C}$ comme suit :

$$(d^0_\Gamma, d^1_\Gamma) \quad : \quad \Gamma_* \to \Gamma \times \Gamma \tag{0.2.3}$$

et tout morphisme respecte ces relations.

Alors le foncteur d'oubli des catégories paramétriques vers les catégories a un adjoint à droite, qui envoie $\mathcal{C}$ sur la catégorie des objets semi-cubiques (c'est-à-dire cubiques avec seulement des faces) dans $\mathcal{C}$. Dans ce cas, l'endofoncteur envoie un objet semi-cubique Γ sur l'objet semi-cubique Γ_* des chemins dans Γ, avec d^0_Γ (resp. d^1_Γ) le morphisme qui envoie un chemin sur sa source (resp. cible).

On argumentera également que les modèles colibrement paramétriques tendent à exister. Pour cela on prouvera l'existence de nombreux adjoints à droite à des foncteurs d'oubli, construisant non seulement des structures cubiques, mais aussi des structures basées sur d'autres formes similaires, ou même des structures apparemment sans rapport.

Remarque 0.2.2. Tous les foncteurs oubliant la paramétricité ont des adjoints à gauche. Ceux-ci permettent de construire des modèles librement paramétriques, qui sont très différents des modèles colibrement paramétriques :

- Le modèle librement paramétrique généré par $\mathcal{C}$ suppose simplement que $\mathcal{C}$ est paramétrique. Ce processus brutal conduit à un modèle incohérent si $\mathcal{C}$ possède un terme contredisant la paramétricité.
- Le modèle colibrement paramétrique généré par $\mathcal{C}$ contient le fragment paramétrique de $\mathcal{C}$. Lorsque $\mathcal{C}$ possède un terme contredisant la paramétricité, il n'apparaîtra pas dans le modèle colibrement paramétrique, sans générer d'incohérence.

0.3. Variantes de la paramétricité

On donne un aperçu des différentes variantes de paramétricité et de leurs rôles dans cette thèse.

- **Arité.** On a pour l'instant supposé que tout type est équipé d'une relation binaire, mais on peut aussi supposer un prédicat (donnant des simplexes augmentés plutôt que des cubes) ou plus généralement une relation n-aire. Cela ne pose aucun problème, et on utilisera la paramétricité unaire lorsque cela conduit à des notations plus légères.

- **Externe / interne.** La paramétricité standard est appelée externe, car elle ne peut pas être utilisée dans la théorie des types. En effet, on ne peut pas prouver que les variables sont reliées à elles-mêmes, de sorte qu'on ne peut pas prouver que les termes ouverts sont reliés à eux-mêmes. Cela empêche d'utiliser la paramétricité sans changer le contexte où on l'utilise. La paramétricité interne tente de corriger cela en introduisant des réflexivités, c'est-à-dire des témoins que toute variable dans un type est reliée à elle-même. Dans ce cas, l'adjoint à droite donne des cubes, avec des faces et des réflexivités.

  Remarque 0.3.1. Les réflexivités ne peuvent être combinées avec les types de fonction ou un univers $\mathcal{U}$ dans notre cadre, car on ne sait pas définir la réflexivité pour :

$$A \to B \tag{0.3.1}$$

inductivement à partir des réflexivités pour A et B, ou bien définir la réflexivité pour une variable de type dans l'univers.

- **Itérée / tronquée.** La paramétricité standard est itérée, dans le sens où elle peut être appliquée autant de fois que l'on veut, construisant ainsi une structure cubique. Il est également possible de définir des variantes tronquées de la paramétricité, qui ne peuvent être appliquées qu'un nombre fixé de fois. Par exemple, les graphes (c'est-à-dire une structure cubique tronquée de dimension 1) sont obtenus à partir d'une paramétricité qui ne peut être appliquée qu'une seule fois. Ces notions tronquées de paramétricité sont traitées explicitement dans [**AGJ14**] et [**GFO16**].
- **Relationnelle / univalente.** On appelle la paramétricité standard relationnelle car elle affirme l'existence de relations. Les variantes affirmant l'existence d'équivalences (c'est-à-dire des relations avec transports) sont appelées univalentes. Elles devraient conduire à des adjoints à droite construisant des structures cubiques de Kan. Ce genre de paramétricité a été considéré dans [**AK17**], et récemment développé en collaboration avec Michael Shulman. Dans [**CH20**], la paramétricité relationnelle et l'univalence sont introduites de manière parallèle, en soulignant les similitudes entre ces deux notions. Ces approches ne s'intègrent pas dans notre cadre car elles utilisent des réflexivités et des types de fonction. Néanmoins, elles constituent des motivations clés dans notre étude de la paramétricité. Voir [**TTS21**] pour une approche alternative supposant un univers univalent au départ, qui semble convenir à notre cadre.

Notre objectif est de prouver que des modèles colibrement paramétriques existent pour de telles variantes. De plus, on veut donner des descriptions explicites pour les adjoints à droite lorsque cela est possible. Comme modèles de la théorie des types, on considèrera principalement les catégories avec familles [**Dyb95**] (voir [**CCD21**] pour une introduction plus récente) et les clans [**Joy17**]. On ne prévoit pas d'obstacle majeur à l'extension de cette thèse à d'autres notions essentiellement algébriques de modèles de la théorie des types, du moins en l'absence de types de fonction et d'univers.

0.4. Les notions de paramétricité comme extensions par section

On suppose donnée une théorie T des modèles de la théorie des types (par exemple T est la théorie des catégories avec familles). Ici, par théorie, on entend une signature pour les types inductifs-inductifs quotients [**KKA19**] (voir [**AK16**] pour un compte rendu plus ancien axé sur les modèles de la théorie des types). On pourrait aussi utiliser une théorie essentiellement algébrique [**AHR99**] ou une théorie algébrique généralisée [**Car86**]. La théorie T' des modèles paramétriques est une extension de T d'une forme très spécifique. On axiomatise cette forme sous le nom d'*extension par section.*

Définition 0.4.1. Une extension par section d'une théorie T est une extension par :

- Des opérations unaires avec des équations les définissant inductivement.

- Des équations unaires prouvables inductivement.

La théorie des modèles paramétriques est donc une extension par section de la théorie des modèles de la théorie des types. On donne une définition précise dans le texte principal.

Remarque 0.4.2. On a introduit les extensions par section dans [**Moe21**] sous le nom relativement générique d'*interprétations*. Dans le chapitre 2 de cette thèse, on donne essentiellement les mêmes résultats que dans [**Moe21**], mais avec de nouvelles définitions ainsi que de nouvelles preuves. Les exemples se recoupent largement entre ces deux sources.

Le résultat principal du chapitre 2 est le suivant :

Théorème 0.4.1. *Soit T' une extension par section de T. Alors le foncteur d'oubli :*

$$U \quad : \quad \mathrm{Alg}_{T'} \to \mathrm{Alg}_T \tag{0.4.1}$$

a un adjoint à droite.

Dans [**Moe21**], l'existence de cet adjoint à droite est prouvée à l'aide de la théorie des catégories localement présentables [**AR94**]. La preuve simplifiée donnée dans cette thèse utilise une définition directe des objets colibres comme limites.

Les principaux exemples d'extensions par section sont la paramétricité standard pour les catégories, les clans et les catégories avec familles, avec ou sans types de fonction et univers.

0.5. Les notions de paramétricité comme modèles monoïdaux

Dans le chapitre 3, on présente une approche alternative, qui donne des descriptions plus compactes pour les modèles colibrement et librement paramétriques. Cela permettra de prouver que de nombreux modèles cubiques sont colibrement paramétriques.

Pour toute théorie T, la théorie des T-algèbres équipées d'un endomorphisme est une extension par section de T. Inspiré par cet exemple, on considère les extensions par une structure de module.

On suppose une catégorie monoïdale symétrique fermée $\mathcal{V}$ de modèles de la théorie des types, et pour $\mathcal{C}$ et $\mathcal{D}$ dans $\mathcal{V}$ on dénote par :

$$\mathcal{C} \multimap \mathcal{D} \tag{0.5.1}$$

l'exponentielle de $\mathcal{D}$ par $\mathcal{C}$. On définit alors une notion de paramétricité pour $\mathcal{V}$ comme un monoïde $\mathcal{M}$ dans $\mathcal{V}$. Un modèle $\mathcal{M}$-paramétrique est défini comme un $\mathcal{M}$-module dans $\mathcal{V}$. On peut alors construire des objets librement et colibrement paramétriques simplement comme des modules induits et coinduits, de sorte qu'on a une chaîne de foncteurs adjoints :

$$\begin{tikzcd} \mathcal{V} \arrow[r, bend left, "\mathcal{C} \mapsto \mathcal{M}\otimes\mathcal{C}"] \arrow[r, bend right, "\mathcal{C} \mapsto \mathcal{M}\multimap\mathcal{C}"'] & \{\mathcal{M}\text{-modules}\} \arrow[l] \end{tikzcd}$$

où la structure de $\mathcal{M}$-module sur $\mathcal{M} \otimes \mathcal{C}$ (resp. $\mathcal{M} \multimap \mathcal{C}$) est induite par la structure de $\mathcal{M}$-module à gauche (resp. à droite) de $\mathcal{M}$ induite par la multiplication.

Exemple 0.5.1. Soit $\mathcal{V}$ la catégorie cartésienne fermée des catégories. Toute catégorie de cubes $\square$ dans [**BM17**] est monoïdale. Elle peut avoir des diagonales, des réflexivités, des symétries, des connexions ou des inverses. Alors $\mathcal{C}^{\square}$ est une catégorie colibrement $\square$-paramétrique pour toute catégorie $\mathcal{C}$.

Par exemple, la paramétricité standard est obtenue en considérant comme $\square$ la catégorie monoïdale librement générée par un objet $\mathbb{I}$ avec deux morphismes :

$$d^0, d^1 \quad : \quad \mathbb{I} \to 1 \tag{0.5.2}$$

Cette catégorie est équivalente à (l'opposé de) la catégorie des semi-cubes.

Exemple 0.5.2. Dans la Section 3.4, on considèrera une variante stricte des catégories exactes à gauche qu'on appellera les catégories lex strictes. Soit $\mathcal{V}$ la catégorie des catégories lex strictes. Dans ce cas, une notion de paramétricité (c'est-à-dire un monoïde dans $\mathcal{V}$) est une catégorie lex stricte avec un produit monoïdal qui préserve les limites finies en chaque variable.

A titre d'exemple, on prouvera que la catégorie lex stricte librement générée par :

$$s, t \quad : \quad E \to V \tag{0.5.3}$$

est en fait une notion de paramétricité où :

- L'objet V est l'unité du produit monoïdal.
- Le produit monoïdal $E \otimes E$ est isomorphe au produit fibré de quatre copies de E dessinant un carré.

Les catégories lex strictes de graphes (c'est-à-dire d'objets cubiques tronqués de dimension 1) sont colibrement paramétriques pour cette notion de paramétricité. Ceci sera étendu aux semi-cubes et aux cubes munis de réflexivités tronqués de dimension n.

Exemple 0.5.3. Dans la Section 3.7, on considèrera une variante stricte des clans qu'on appellera les clans stricts. Soit $\mathcal{V}$ la catégorie des clans stricts. Un monoïde dans $\mathcal{V}$ sera un clan strict muni d'un produit monoïdal :

- Qui préservent les limites en chaque variable.
- Tel que, étant données deux fibrations :

$$i \quad \twoheadrightarrow \quad i' \tag{0.5.4}$$
$$j \quad \twoheadrightarrow \quad j' \tag{0.5.5}$$

  on a une fibration induite :

$$i \otimes j \quad \twoheadrightarrow \quad (i' \otimes j) \underset{i' \otimes j'}{\times} (i \otimes j') \tag{0.5.6}$$

À titre d'exemple, on prouvera que les clans de semi-cubes fibrants au sens de Reedy sont colibrement paramétriques, en utilisant comme notion de paramétricité le clan strict monoïdal librement généré par un objet $\mathbb{I}$ et une fibration :

$$\mathbb{I} \quad \twoheadrightarrow \quad 1 \times 1 \tag{0.5.7}$$

On prouvera aussi le résultat analogue pour les cubes munis de réflexivités.

0.6. Plan

Cette thèse est divisée en trois chapitres.

Le chapitre 1 est une introduction à la paramétricité et aux modèles de la théorie des types :

- Dans les sections 1.1 à 1.3, la paramétricité standard, où tout type est équipé d'une relation, est introduite pour les catégories et les clans. Ceci présente le point de vue catégorique sur la paramétricité utilisé dans cette thèse.
  - Une introduction aux clans est donnée dans la section 1.2.
  - Quelques exemples de modèles paramétriques inspirés de la théorie de l'homotopie sont donnés dans la section 1.1 pour les catégories et la section 1.3 pour les clans, soulignant la pertinence de la paramétricité.
- Une introduction aux catégories avec familles est donnée dans la section 1.4. Les catégories avec familles paramétriques ne sont pas définies ici.
- Les principales variantes de la paramétricité sont passées en revue dans la section 1.5, avec des références à la littérature. Cette section contient du vocabulaire utilisé dans le reste de la thèse.

Le chapitre 2 traite des extensions par section :

- La section 2.1 présente les extensions par section catégoriques, qui sont des foncteurs d'oubli d'une forme particulière. On prouve qu'elles ont des adjoints à droite.
- Les signatures pour les types inductifs-inductifs quotients sont présentées dans la section 2.2, avec des références à [**KKA19**] et [**KK20**]. Ces signatures sont ensuite utilisées pour définir les extensions par section.
- Section 2.3 contient le résultat principal de ce chapitre, à savoir que le foncteur d'oubli associé à une extension par section a un adjoint à droite. Ce résultat a déjà été prouvé dans [**Moe21**], mais on donne ici une nouvelle preuve, à l'aide des extensions par section catégoriques.
- Dans les sections 2.4 à 2.7, on prouve que les notions standard de paramétricité pour les catégories, les clans et les catégories avec familles (avec ou sans types de fonction et un univers) sont effectivement des extensions par section. Les parties les plus techniques des preuves pour les catégories avec familles peuvent être trouvées dans l'appendice A.
- Dans la section 2.8, on esquisse deux exemples conjecturaux d'extensions par section, tous deux concernant des formes restreintes d'univalence. D'abord, on explique pourquoi l'on s'attend à ce que la *théorie des types de setoïdale* [**ABKT19**] modélise une forme de paramétricité univalente tronquée à la dimension 1. On doit pour cela exclure les types de fonction et l'univers pour les ensembles. On considère ensuite la *paramétricité univalente* de [**TTS21**].

Le chapitre 3 traite des modèles paramétriques définis comme modules :

- Les sections 3.1 à 3.3 axiomatisent les notions de paramétricité comme monoïdes, et prouvent que les modèles colibrement paramétriques existent dans ce contexte.
  - La section 3.1 sert de motivation. Elle explique comment les catégories paramétriques de la section 1.1 peuvent être vues comme des modules

sur une catégorie monoïdale. Cet exemple guide les deux sections suivantes.
  - Les notions de paramétricité et de modèles paramétriques sont définies comme des monoïdes et des modules dans la section 3.2. Les définitions de monoïde et de module dans une catégorie monoïdale quelconque sont rappelées.
  - Dans la section 3.3, on prouve que les foncteurs oubliant une structure de module ont des adjoints à gauche et à droite. La preuve donnée utilise l'interprétation du fragment multiplicatif de la logique linéaire dans les catégories monoidales symétriques fermées, bien qu'une preuve directe soit possible.
- Cette axiomatisation est utilisée pour les catégories lex strictes dans les sections 3.4 à 3.6.
  - Dans la section 3.4, on prouve que les catégories lex strictes forment une catégorie monoïdale symétrique fermée. Il est nécessaire de considérer une version stricte des catégories lex pour que ce résultat soit vrai. Plus précisément on considère que les morphismes entre catégories lex strictes commutent avec les limites à égalité près, et on suppose que les isomorphismes canoniques de commutation des limites dans une catégorie lex stricte sont des identités. Les monoïdes dans cette catégorie des catégories lex strictes (aussi appelés catégories lex monoïdales strictes) sont explicités dans la section 3.5.
  - Des notions tronquées de paramétricité pour les catégories lex strictes sont définies dans la section 3.6. Plus précisément, on prouve que les catégories lex strictes d'objets semi-cubiques (ou cubiques avec réflexivités) tronqués de dimension n sont colibrement paramétriques.
- Cette axiomatisation est utilisée pour les clans dans les sections 3.7 à 3.9.
  - Les sections 3.7 et 3.8 contiennent une adaptation des sections 3.4 et 3.5 aux clans. Ainsi, toutes les définitions et preuves sont étendues aux fibrations, ce qui donne une catégorie monoïdale symétrique fermée des clans stricts.
  - Dans la section 3.9, on prouve que les clans d'objets semi-cubiques (ou cubiques avec réflexivités) fibrants au sens de Reedy sont colibrement paramétriques.

CHAPTER 1

Parametric models of type theory

In this thesis we will study type theory from a semantical point of view, so that we do not define type theory itself but rather structures where it can be interpreted. These structures are called models of type theory. From this point of view, the syntax of type theory corresponds to the initial model, and interpretations of type theory correspond to morphisms out of this initial model.

In this chapter, we will introduce two notions of model for type theory: categories with families and clans. These notions are formally similar (although not equivalent), but have different goals:

- The notion of category with families was introduced by Peter Dybjer [**Dyb95, CCD21**] as a notion of model of type theory very close to syntax. It is therefore accessible to syntactically-minded type theorists.
- The notion of clan was designed by Joyal to help bridge the gap between type theory and homotopy theory [**Joy17**]. It should appear reasonable to homotopically-minded category theorists.

Our goal here is to introduce these two notions of model of type theory, as well as parametricity for clans. Parametricity for categories with families is defined later, in Section 2.6. This chapter is organized as follows:

- In Section 1.1 we introduce parametricity for categories. There are interesting examples already in this simple case.
- In Section 1.2 we define clans. They are models of type theory where types over a context Γ are represented by fibrations with target Γ.
- In Section 1.3 we define parametric clans and give examples.
- In Section 1.4 we define categories with families. They are close to the syntax of type theory, so it can be hard to build semantically-flavored examples.
- In Section 1.5 we give an informal overview of variants of parametricity, with pointers to the literature. This thesis aims to develop a rigorous framework to study these variants, and techniques to prove theorems about them.

1.1. Parametricity for categories

A category can be seen as a rudimentary model of type theory, with objects as types and morphisms as terms. We give three equivalent definitions for parametric categories. Recall that a model is called parametric if any type comes with a relation, and any term preserves these. We apply this intuition as directly as possible in the following definition:

Definition 1.1.1. A parametric category is a category $\mathcal{C}$ equipped with:

- For any object Γ, a relation internal to $\mathcal{C}$, i.e. an object Γ_* with:

$$d^0_\Gamma, d^1_\Gamma \quad : \quad \mathrm{Hom}_{\mathcal{C}}(\Gamma_*, \Gamma) \tag{1.1.1}$$

- For any morphism:

$$\sigma \quad : \quad \mathrm{Hom}_{\mathcal{C}}(\Gamma, \Delta) \tag{1.1.2}$$

we have:

$$\sigma_* \quad : \quad \mathrm{Hom}_{\mathcal{C}}(\Gamma_*, \Delta_*) \tag{1.1.3}$$

with commutative squares:

$$\begin{tikzcd} \Gamma_* \arrow[r, "\sigma_*"] \arrow[d, "d^\epsilon_\Gamma"'] & \Delta_* \arrow[d, "d^\epsilon_\Delta"] \\ \Gamma \arrow[r, "\sigma"'] & \Delta \end{tikzcd}$$

for $\epsilon = 0$ or 1.
- Moreover $\_{}_*$ respects composition and identity:

$$(\sigma \circ \delta)_* \quad = \quad \sigma_* \circ \delta_* \tag{1.1.4}$$

$$(\mathrm{id}_\Gamma)_* \quad = \quad \mathrm{id}_{\Gamma_*} \tag{1.1.5}$$

Being parametric is a structure on a category rather than a property. This means that a category can be parametric in several different ways. We give an alternative definition, formulated in the usual categorical language.

Definition 1.1.2. A parametric category is a category $\mathcal{C}$ equipped with:

- An endofunctor of $\mathcal{C}$ denoted by:

$$\Gamma \quad \mapsto \quad \Gamma_* \tag{1.1.6}$$

- For any $\Gamma : \mathcal{C}$ two morphisms:

$$d^0_\Gamma, d^1_\Gamma \quad : \quad \mathrm{Hom}(\Gamma_*, \Gamma) \tag{1.1.7}$$

natural in Γ.

Another alternative definition can be given using the category $\mathcal{G}$ freely generated by:

$$E \rightrightarrows V \tag{1.1.8}$$

A functor from $\mathcal{G}$ to a category $\mathcal{C}$ is an object in $\mathcal{C}$ with a relation internal to $\mathcal{C}$, i.e. a graph in $\mathcal{C}$.

Definition 1.1.3. A parametric category is a category $\mathcal{C}$ equipped with a section of the evaluation functor at V:

$$\mathrm{ev}_V \quad : \quad \mathcal{C}^{\mathcal{G}} \to \mathcal{C} \tag{1.1.9}$$

Remark 1.1.4. Given an object Γ in a parametric category $\mathcal{C}$, we can iterate $\_{}_*$ and d^0, d^1, building the following diagram:

$$
\begin{tikzcd}
\Gamma & \Gamma_* \arrow[l, shift right, "d^0_\Gamma"'] \arrow[l, shift left, "d^1_\Gamma"] & \Gamma_{**} \arrow[l, shift right, "d^0_{\Gamma_*}"'] \arrow[l, shift left, "d^1_{\Gamma_*}"] \arrow[l, bend right=60, "(d^0_\Gamma)_*"'] \arrow[l, bend left=60, "(d^1_\Gamma)_*"] & \cdots
\end{tikzcd}
$$

This will turn out to be a semi-cubical object in $\mathcal{C}$ (meaning a cubical object without reflexivities) with Γ as its object of points, as we will see in Remark 3.1.10.

We give examples of parametricity structures:

Example 1.1.5. For any category $\mathcal{C}$ we can define:

$$\Gamma_* = \Gamma \tag{1.1.10}$$
$$d^0_\Gamma(x) = x \tag{1.1.11}$$
$$d^1_\Gamma(x) = x \tag{1.1.12}$$

Here any type comes with its equality relation.

Example 1.1.6. If the category $\mathcal{C}$ has products, we can define:

$$\Gamma_* = \Gamma \times \Gamma \tag{1.1.13}$$
$$d^0_\Gamma(x, y) = x \tag{1.1.14}$$
$$d^1_\Gamma(x, y) = y \tag{1.1.15}$$

Here any type comes with the trivial (i.e. always true) relation.

Example 1.1.7. If $\mathcal{C}$ has an initial object $\perp$, we can define:

$$\Gamma_* = \perp \tag{1.1.16}$$

This uniquely determines d^0_Γ and d^1_Γ. Here any type comes with the empty (i.e. always false) relation.

In all these examples the maps:

$$(d^0_\Gamma, d^1_\Gamma) \quad : \quad \Gamma_* \to \Gamma \times \Gamma \tag{1.1.17}$$

are monomorphisms, so that they correspond to relations in the proof-irrelevant sense, meaning that elements can be related in at most one way.

Remark 1.1.8. Assuming the law of excluded middle, there is no other such proof-irrelevant parametricity on the category of sets. Indeed:

- Given two distinct related elements, say $x, x' \in X$, then for any $y, y' \in Y$ we have a map:

  $$f \quad : \quad X \to Y \tag{1.1.18}$$

  such that:

  $$f(x) = y \tag{1.1.19}$$
  $$f(x') = y' \tag{1.1.20}$$

  But any map sends related inputs to related outputs, so that y and y' are related.

- Similarly, given an element related to itself, say $x \in X$, then for any $y \in Y$ we have a map:

$$f \quad : \quad X \to Y \tag{1.1.21}$$

sending x to y, so that y is related to itself.

From this we know that given a proof-irrelevant parametricity on sets:

- If two distinct elements are related, all elements are related and we have the trivial relation.
- If no distinct elements are related, but an element is related to itself then all elements are and we have the equality relation.
- Otherwise no elements are related and we have the empty relation.

Now we give examples of parametricity structures with proof-relevant relations.

Example 1.1.9. In any cartesian closed category of spaces containing the unit interval:

$$[0,1] \subset \mathbb{R} \tag{1.1.22}$$

we can define:

$$\Gamma_* \quad = \quad \Gamma^{[0,1]} \tag{1.1.23}$$

$$d^0_\Gamma(p) \quad = \quad p(0) \tag{1.1.24}$$

$$d^1_\Gamma(p) \quad = \quad p(1) \tag{1.1.25}$$

For any points x and y in Γ, the fiber of the map:

$$(d^0_\Gamma, d^1_\Gamma) \quad : \quad \Gamma^{[0,1]} \to \Gamma \times \Gamma \tag{1.1.26}$$

over (x, y) is the space of paths from x to y in Γ.

Example 1.1.10. We can extend Example 1.1.9 to any monoidal closed category $\mathcal{C}$ with an object I and two maps:

$$e_0, e_1 \quad : \quad 1 \to I \tag{1.1.27}$$

where 1 is the unit. Indeed, writing:

$$\_ \multimap \_ \tag{1.1.28}$$

for the exponential in $\mathcal{C}$, we can define a parametricity by:

$$\Gamma_* \quad = \quad I \multimap \Gamma \tag{1.1.29}$$

$$d^0_\Gamma \quad = \quad \eta_\Gamma \circ (e_0 \multimap \Gamma) \tag{1.1.30}$$

$$d^1_\Gamma \quad = \quad \eta_\Gamma \circ (e_1 \multimap \Gamma) \tag{1.1.31}$$

where:

$$\eta_\Gamma \quad : \quad (1 \multimap \Gamma) \cong \Gamma \tag{1.1.32}$$

is induced by the monoidal closed structure.

Example 1.1.11. We can also build a parametricity on any category with products and a functorial factorization system. Indeed we can define a parametricity by factoring the diagonal of any object Γ as follows:

$$\Gamma \to \Gamma_* \xrightarrow{(d^0_\Gamma, d^1_\Gamma)} \Gamma \times \Gamma \tag{1.1.33}$$

Example 1.1.12. We will see in Example 3.3.2 that for any category $\mathcal{C}$, the category of semi-cubical objects in $\mathcal{C}$ is parametric. For a semi-cubical object Γ in $\mathcal{C}$, the object Γ_* is the semi-cubical object of paths in Γ. Then d^0_Γ (resp. d^1_Γ) sends a path to its source (resp. target).

This holds for all kinds of cubical objects.

We also have examples using low-dimensional homotopically-flavored objects, e.g. categories.

Example 1.1.13. We can define a parametricity on the category of categories by:

$$\begin{aligned} \Gamma_* &= (x_0, x_1 : \Gamma) \times \mathrm{Hom}_\Gamma(x_0, x_1) && (1.1.34) \\ d^0_\Gamma(x_0, x_1, f) &= x_0 && (1.1.35) \\ d^1_\Gamma(x_0, x_1, f) &= x_1 && (1.1.36) \end{aligned}$$

Here two objects in a category are related if we have a morphism between them.

Example 1.1.14. We can define a parametricity on the category of categories by:

$$\begin{aligned} \Gamma_* &= (x_0, x_1, r : \Gamma) \times \mathrm{Hom}_\Gamma(r, x_0) \times \mathrm{Hom}_\Gamma(r, x_1) && (1.1.37) \\ d^0_\Gamma(x_0, x_1, \cdots) &= x_0 && (1.1.38) \\ d^1_\Gamma(x_0, x_1, \cdots) &= x_1 && (1.1.39) \end{aligned}$$

Here two objects in a category $\mathcal{C}$ are related if there is a relation internal to $\mathcal{C}$ between them.

To summarize this section, the definition of parametric category is both natural and meaningful:

- Natural as it can be compactly stated using category theory.
- Meaningful as many homotopically-flavored structures on a category imply that they are parametric.

1.2. Clans

A clan is a model of type theory where types are modeled by fibrations. The intuition is as follows:

- A fibration:

$$f \quad : \quad A \to \Gamma \qquad (1.2.1)$$

  is a map with fibers varying continuously in Γ.
- A dependent type over Γ is a family of types varying continuously in Γ.

So a dependent type over a context Γ is modeled by a fibration with target Γ.

Remark 1.2.1. Axiomatising the notion of fibration has been done in many different ways, notably via model categories [**Qui67**] and categories of fibrant objects [**Bro73**]. Fibrations in clans retain very little features from these classical homotopical axiomatisations.

First we give an auxiliary definition.

Definition 1.2.2. A class of maps $\mathcal{F}$ in a category $\mathcal{C}$ is called stable under pullback if for any diagram:

$$\begin{tikzcd} & \Delta \arrow[d, "\sigma"] \\ A \arrow[r, "p"'] & \Gamma \end{tikzcd}$$

with p in $\mathcal{F}$, there exists a pullback square:

$$\begin{tikzcd} A \underset{\Gamma}{\times} \Delta \arrow[r, "\pi_2"] \arrow[d] & \Delta \arrow[d, "\sigma"] \\ A \arrow[r, "p"'] & \Gamma \end{tikzcd}$$

where π_2 is in $\mathcal{F}$.

For any commutative square:

$$\begin{tikzcd} B \arrow[r, "\theta"] \arrow[d, "\delta"'] & \Delta \arrow[d, "\sigma"] \\ A \arrow[r, "p"'] & \Gamma \end{tikzcd}$$

we denote the induced map by:

$$(\delta, \theta) \quad : \quad B \to A \underset{\Gamma}{\times} \Delta \tag{1.2.2}$$

Recall that fibrations are intuitively maps with continuously varying fibers.

Definition 1.2.3. A clan is a category with a terminal object, together with a class of maps called fibrations such that:

- Fibrations are stable under isomorphism, composition and pullback.
- Maps to the terminal object are fibrations.

For now we use a weak notion of morphism between clans.

Definition 1.2.4. A morphism between clans is a functor preserving fibrations, terminal objects and pullbacks of fibrations up to isomorphisms.

Remark 1.2.5. Latter in the thesis we will be forced to consider functors commuting with limits on the nose, to avoid considering a 2-category of clans.

We write:

$$p \quad : \quad A \twoheadrightarrow \Gamma \tag{1.2.3}$$

when p is a fibration. Clans are not required to have all finite limits, as pullbacks along non-fibrations do not necessarily exist.

Remark 1.2.6. Any identity map id_A is a fibration. Indeed it is isomorphic to the pullback of $\mathrm{id}_\top$ along the unique map:

$$\epsilon_A \quad : \quad A \to \top \tag{1.2.4}$$

where $\top$ is the terminal object.

Remark 1.2.7. Cartesian products are defined in any clan as pullbacks:

$$\begin{tikzcd} A \times B \arrow[r, "\pi_2"] \arrow[d, "\pi_1"'] & B \arrow[d] \\ A \arrow[r] & \top \end{tikzcd}$$

We see that cartesian projections are always fibrations.

Remark 1.2.8. We give the correspondence between assumptions on fibrations and types:

Fibrations	Types
Pullbacks	Substitutions
Identities	Unit types
Compositions	Product types
Maps to $\top$	Democracy
Projections	Constant types

The product types in this thesis are often called Σ-types in other sources. See Definition 1.4.11 for more on democracy.

Example 1.2.9. The following are minimal and maximal examples of clans:

- Any cartesian category (i.e. category with finite products) with maps isomorphic to projections as fibrations.
- Any lex category (i.e. category with finite limits) with all maps as fibrations.

Example 1.2.10. The category of fibrant objects in any model category is a clan. Many clans are not of this form.

We will use model categories for a few examples in the rest of this chapter. A thorough treatment can be found in [**Hov07**], and a shorter introduction in Appendix A.2 of [**Lur09**].

1.3. Parametricity for clans

Now we extend parametricity from categories to clans. A parametric clan is essentially a clan with path spaces.

Definition 1.3.1. A parametric clan is a parametric category $\mathcal{C}$ equipped with a clan structure such that:

- The endofunctor:
$$\Gamma \quad \mapsto \quad \Gamma_* \tag{1.3.1}$$
of $\mathcal{C}$ is a clan morphism.
- For any fibration:
$$p \quad : \quad A \twoheadrightarrow \Gamma \tag{1.3.2}$$
in $\mathcal{C}$, the commutative square:
$$\begin{tikzcd} A_* \arrow[r, "p_*"] \arrow[d, "(d^0_A, d^1_A)"'] & \Gamma_* \arrow[d, "(d^0_\Gamma, d^1_\Gamma)"] \\ A \times A \arrow[r, "p \times p"'] & \Gamma \times \Gamma \end{tikzcd}$$

induces a fibration:

$$A_* \twoheadrightarrow (A \times A) \underset{\Gamma\times\Gamma}{\times} \Gamma_* \tag{1.3.3}$$

We can prove that fibrations are stable under product, so that $p \times p$ is a fibration and the pullback in the previous definition exists.

Remark 1.3.2. Let $\mathcal{C}$ be a parametric clan and Γ be an object in $\mathcal{C}$. We have an induced fibration:

$$(d^0_\Gamma, d^1_\Gamma) \quad : \quad \Gamma_* \to \Gamma \times \Gamma \tag{1.3.4}$$

Indeed, from the fibration:

$$\epsilon_\Gamma \quad : \quad \Gamma \twoheadrightarrow \top \tag{1.3.5}$$

we get an induced fibration:

$$((d^0_\Gamma, d^1_\Gamma), (\epsilon_\Gamma)_*) \quad : \quad \Gamma_* \twoheadrightarrow (\Gamma \times \Gamma) \underset{\top\times\top}{\times} \top_* \tag{1.3.6}$$

But $\__*$ is a morphism of clans so that $\top_*$ is a terminal object, so this map is isomorphic to (d^0_Γ, d^1_Γ).

Remark 1.3.3. Interpreting fibrations as types, Remark 1.3.2 means that in a parametric clan any context Γ comes with a relation:

$$\Gamma_0, \Gamma_1 \vdash \Gamma_* \tag{1.3.7}$$

where Γ_0 and Γ_1 are two copies of Γ.

Then the condition on fibrations in the definition of parametricity says that any type $\Gamma \vdash A$ comes with a type:

$$\Gamma_0, \Gamma_1, \Gamma_*, A_0, A_1 \vdash A_* \tag{1.3.8}$$

where A_0 (resp. A_1) is a copy of A depending on Γ_0 (resp. Γ_1).

Remark 1.3.4. Consider $\mathcal{G}$ the free clan generated by:

$$E \twoheadrightarrow V \times V \tag{1.3.9}$$

In Definition 3.8.1 we will give an exponential:

$$\_ \multimap \_ \tag{1.3.10}$$

for clans such that a clan is parametric when we have a section of the evaluation functor at V:

$$\mathrm{ev}_V \quad : \quad (\mathcal{G} \multimap \mathcal{C}) \to \mathcal{C} \tag{1.3.11}$$

Lemma 1.3.5. *Assume given a monoidal model category $\mathcal{C}$ with 1 cofibrant, then its category of fibrant objects is a parametric clan.*

Proof. We assume some familiarity with monoidal model categories. We factor the codiagonal of 1 as:

$$(1+1) \rightarrowtail I \xrightarrow{\sim} \!\!\!\!\rightarrow 1$$

For any object Γ, by exponentiating the cofibration:

$$(1+1) \rightarrowtail I \tag{1.3.12}$$

we get a fibration:

$$(I \multimap \Gamma) \twoheadrightarrow \Gamma \times \Gamma$$

So we define:

$$\Gamma_* \quad = \quad I \multimap \Gamma \tag{1.3.13}$$

as in Example 1.1.10. To check that this is a parametricity, we need to check that:

- The functor $\_{}_*$ is a morphism of clan. Indeed:
  - It is the right adjoint of the functor:
    $$\Gamma \mapsto \Gamma \otimes I \tag{1.3.14}$$
    so that it commutes with limits.
  - The unit 1 is assumed cofibrant, so that $1 + 1$ and therefore I are cofibrant as well. This implies that $\_{}_*$ preserves fibrations.
- We have a cofibration:
  $$(1 + 1) \rightarrowtail I \tag{1.3.15}$$
  so that for any fibration:
  $$A \twoheadrightarrow \Gamma \tag{1.3.16}$$
  we have an induced fibration:
  $$(I \multimap A) \quad \twoheadrightarrow \quad (1 + 1 \multimap A) \underset{1+1\multimap\Gamma}{\times} (I \multimap \Gamma) \tag{1.3.17}$$
  However, this map is isomorphic to:
  $$A_* \quad \longrightarrow \quad (A \times A) \underset{\Gamma\times\Gamma}{\times} \Gamma_* \tag{1.3.18}$$
  giving the required condition.

□

Example 1.3.6. We can build many parametric clans using the previous lemma, for example:

- The category of compactly generated spaces with Serre fibrations.
- The category of Kan simplicial sets with Kan fibrations.
- The category of compactly generated pointed spaces with Serre fibrations.
- The category of chain complexes with projective fibrations.

Example 1.3.7. Let $\mathcal{C}$ be a clan. We will see in Section 3.9 that the clan of Reedy fibrant semi-cubical (or cubical with reflexivities) objects in $\mathcal{C}$ is parametric.

1.4. Categories with families

In Section 1.2 we presented clans as an homotopically-flavored notion of model of type theory. In this section, we introduce the notion of category with families [**Dyb95**] as a syntactically-flavored alternative.

We adopt a categorical presentation following the trend in this chapter. In Section 2.6 we will give an alternative definition of categories with families as algebras for a Quotient Inductive-Inductive Type (abbreviated as QIIT) signature, making clearer the connection with syntax.

First we define families.

Definition 1.4.1. A family is a set A with a set B_x for any $x \in A$. It is written $(B_x)_{x\in A}$.

Note that families form a category.

Definition 1.4.2. A morphism from the family $(B_x)_{x \in A}$ to the family $(B'_y)_{y \in A'}$ consists of:

$$f \quad : \quad A \to A' \tag{1.4.1}$$

with, for all $x \in A$, a map:

$$g_x \quad : \quad B_x \to B'_{f(x)} \tag{1.4.2}$$

Given a context Γ we have a set of types over Γ, and given a type A over a context Γ we have a set of terms of type A. So any context should come with a family of types and terms. The notion of category with families axiomatizes this:

Definition 1.4.3. A category with family is a category $\mathcal{C}$ with:

- A terminal object.
- A functor from $\mathcal{C}^{op}$ to families, sending $\Gamma : \mathcal{C}$ to a family denoted by:

  $$(\mathrm{Tm}(\Gamma, A))_{A \in \mathrm{Ty}(\Gamma)} \tag{1.4.3}$$

  Given:

  $$\sigma \quad : \quad \mathrm{Hom}_{\mathcal{C}}(\Delta, \Gamma) \tag{1.4.4}$$
  $$A \quad : \quad \mathrm{Ty}(\Gamma) \tag{1.4.5}$$
  $$a \quad : \quad \mathrm{Tm}(\Gamma, A) \tag{1.4.6}$$

  we denote by:

  $$A[\sigma] \quad : \quad \mathrm{Ty}(\Delta) \tag{1.4.7}$$
  $$a[\sigma] \quad : \quad \mathrm{Tm}(\Delta, A[\sigma]) \tag{1.4.8}$$

  the images assumed by functoriality.
- Moreover, for any $\Gamma : \mathcal{C}$ and $A : \mathrm{Ty}(\Gamma)$, we assume given a representing object for the functor:

  $$(\mathcal{C}_{/\Gamma})^{op} \quad \to \quad \mathrm{Set} \tag{1.4.9}$$
  $$(\sigma : \mathrm{Hom}_{\mathcal{C}}(\Delta, \Gamma)) \quad \mapsto \quad \mathrm{Tm}(\Delta, A[\sigma]) \tag{1.4.10}$$

  where $\mathcal{C}_{/\Gamma}$ is the category of arrows to Γ in $\mathcal{C}$, with commutative triangles as morphisms.

Objects of a category with families are called contexts, and morphisms are called substitutions.

Remark 1.4.4. Explicitly, the third condition means that given $\Gamma : \mathcal{C}$ and $A : \mathrm{Ty}(\Gamma)$ we have:

$$(\Gamma, A) \quad : \quad \mathcal{C} \tag{1.4.11}$$
$$\mathrm{w}_A \quad : \quad \mathrm{Hom}_{\mathcal{C}}((\Gamma, A), \Gamma) \tag{1.4.12}$$

such that for any:

$$\sigma \quad : \quad \mathrm{Hom}_{\mathcal{C}}(\Delta, \Gamma) \tag{1.4.13}$$

we have a natural isomorphism between:

$$\mathrm{Tm}(\Delta, A[\sigma]) \tag{1.4.14}$$

and the set of:

$$\delta \quad : \quad \mathrm{Hom}_{\mathcal{C}}(\Delta, (\Gamma, A)) \tag{1.4.15}$$

such that:

$$\mathrm{w}_A \circ \delta \quad = \quad \sigma \tag{1.4.16}$$

Remark 1.4.5. We sketch the correspondence between the notion of category with families and the syntactical presentation of type theory.

- An object in $\mathcal{C}$ corresponds to a context, that is a sequence of well-typed variable declarations:
$$x_1 : A_1, \cdots, x_n : A_n \tag{1.4.17}$$
where A_k can depend on:
$$x_1 : A_1, \cdots, x_{k-1} : A_{k-1} \tag{1.4.18}$$
- The terminal object corresponds to the empty declaration of variables.
- A morphism from:
$$x_1 : A_1, \cdots, x_n : A_n \tag{1.4.19}$$
to
$$y_1 : B_1, \cdots, y_m : B_m \tag{1.4.20}$$
is a substitution, that is a sequence of terms:
$$t_1 \quad : \quad B_1 \tag{1.4.21}$$
$$t_2 \quad : \quad B_2[y_1/t_1] \tag{1.4.22}$$
$$\vdots \tag{1.4.23}$$
$$t_n \quad : \quad B_n[y_1/t_1, \cdots, y_{n-1}/t_{n-1}] \tag{1.4.24}$$
depending on $x_1, \cdots, x_n$.
- For any context Γ:
  - The set $\mathrm{Ty}(\Gamma)$ consists of the well-formed types depending on variables declared in Γ.
  - The set $\mathrm{Tm}(\Gamma, A)$ consists of the terms of type A depending on variables declared in Γ.
- Given type $A : \mathrm{Ty}(\Gamma)$ where Γ is:
$$x_1 : A_1, \cdots, x_n : A_n \tag{1.4.25}$$
and a substitution from Δ to Γ given by:
$$t_1 : A_1, \cdots, t_n : A_n \tag{1.4.26}$$
where $t_1, \cdots, t_n$ depend on variables in Δ, then $A[\sigma] : \mathrm{Ty}(\Delta)$ is:
$$A[x_1/t_1, \cdots, x_n/t_n] \tag{1.4.27}$$
and similarly for terms.
- For any type $B : \mathrm{Ty}(\Gamma)$ where Γ is:
$$x_1 : A_1, \cdots, x_n : A_n \tag{1.4.28}$$
the context (Γ, B) is the context:
$$x_1 : A_1, \cdots, x_n : A_n, y : B \tag{1.4.29}$$
This satisfies the correct universal property.

A context in a category with families is not always of the form:

$$x_1 : A_1, \cdots, x_n : A_n \tag{1.4.30}$$

meaning that it is not always built by repeated context extension from the empty context. So the intuitions above are not part of a rigorous equivalence.

Now we define unit types.

Definition 1.4.6. A category with families is said to have unit types if for any context Γ we have a type:

$$\top \quad : \quad \mathrm{Ty}(\Gamma) \tag{1.4.31}$$

with natural isomorphisms:

$$\mathrm{Tm}(\Gamma, \top) \quad \cong \quad \{\mathrm{t}\} \tag{1.4.32}$$

where $\{\mathrm{t}\}$ is a terminal set.

Remark 1.4.7. Naturality here means that:

$$\top[\sigma] \quad = \quad \top \tag{1.4.33}$$

$$\mathrm{t}[\sigma] \quad = \quad \mathrm{t} \tag{1.4.34}$$

for any:

$$\sigma \quad : \quad \mathrm{Hom}_{\mathcal{C}}(\Gamma, \Delta) \tag{1.4.35}$$

Remark 1.4.8. Unit types correspond to identities being fibrations. Indeed we have a commutative triangle:

$$\begin{tikzcd}
\Gamma \arrow[rr, "\mathrm{id}_\Gamma"] \arrow[dr, "\cong"'] & & \Gamma \\
& (\Gamma, \top) \arrow[ur, two heads, "\mathrm{w}_\top"'] &
\end{tikzcd}$$

Now we define product types.

Definition 1.4.9. A category with families is said to have product types if for any context Γ with:

$$A \quad : \quad \mathrm{Ty}(\Gamma) \tag{1.4.36}$$

$$B \quad : \quad \mathrm{Ty}(\Gamma, A) \tag{1.4.37}$$

we have a type:

$$\Sigma(A, B) \quad : \quad \mathrm{Ty}(\Gamma) \tag{1.4.38}$$

with natural isomorphisms:

$$\mathrm{Tm}(\Gamma, \Sigma(A, B)) \quad \cong \quad (a : \mathrm{Tm}(\Gamma, A)) \times \mathrm{Tm}(\Gamma, B[\mathrm{id}, a]) \tag{1.4.39}$$

where:

$$(\mathrm{id}, a) \quad : \quad \mathrm{Hom}_{\mathcal{C}}(\Gamma, (\Gamma, A)) \tag{1.4.40}$$

is defined through the universal property of (Γ, A).

Here naturality means the appropriate commutations with substitutions. Precise rules are given in Section 2.6.

Remark 1.4.10. Product types correspond to the stability of fibrations under composition. Indeed we have a commutative square:

$$\begin{tikzcd}
(\Gamma, A, B) \arrow[r, "\cong"] \arrow[d, two heads, "w_B"'] & (\Gamma, \Sigma(A, B)) \arrow[d, two heads, "w_{\Sigma(A,B)}"] \\
(\Gamma, A) \arrow[r, two heads, "w_A"'] & \Gamma
\end{tikzcd}$$

Now we define democratic categories with families.

Definition 1.4.11. A category with families is democratic if for any context Γ, there is a type in the empty context:

$$A \quad : \quad \mathrm{Ty}(\top) \tag{1.4.41}$$

such that:

$$\Gamma \quad \cong \quad \top, A \tag{1.4.42}$$

In a democratic category with families, any structure assumed on types is automatically present on contexts as well.

Remark 1.4.12. In a clan, democracy corresponds to any map to $\top$ being a fibration. Indeed for Γ a context we have a fibration:

$$\epsilon_\Gamma \quad : \quad \Gamma \twoheadrightarrow \top \tag{1.4.43}$$

corresponding to a type over $\top$. Its context comprehension is the source of ϵ_Γ, that is the context Γ.

It is significantly harder to give examples of categories with families than clans, because of the requirement that:

$$A[\sigma][\delta] \quad = \quad A[\sigma \circ \delta] \tag{1.4.44}$$

with an actual equality rather than an isomorphism. A model obeying this requirement is said to have *strict substitutions*.

Example 1.4.13. There is a category with families of sets.

- Its category of context is the category of (small) sets.
- A type $A : \mathrm{Ty}(\Gamma)$ is defined as a family of (small) sets $(A_x)_{x\in\Gamma}$
- A term in the type $(A_x)_{x\in\Gamma}$ consists of:

$$a_x \in A_x \tag{1.4.45}$$

  for all $x \in \Gamma$.
- Given:

$$\sigma : \mathrm{Hom}_{\mathrm{Set}}(\Delta, \Gamma) \tag{1.4.46}$$

  and a type $(A_x)_{x\in\Gamma}$ over Γ, for $x \in \Delta$ we define:

$$(A[\sigma])_x \quad = \quad A_{\sigma(x)} \tag{1.4.47}$$

- Given a set Γ and a family:

$$(A_x)_{x\in\Gamma} \tag{1.4.48}$$

  of sets indexed by Γ, we define:

$$(\Gamma, A) \quad = \quad \{(x, y) \mid x \in \Gamma, y \in A_x\} \tag{1.4.49}$$

Example 1.4.14. There is a category with families of categories:

- Its category of context is the category of categories.
- A type A:Ty(Γ) is defined as a functor from A to the category of categories.
- Given a category Γ with a functor:

$$A \quad : \quad \Gamma \to \mathrm{Cat} \tag{1.4.50}$$

  we define the category (Γ, A) as follows:
  - Its objects are pairs of:

$$x \quad : \quad \mathrm{Ob}_\Gamma \tag{1.4.51}$$
$$y \quad : \quad \mathrm{Ob}_{A(x)} \tag{1.4.52}$$

  - A morphism from (x, y) to (x', y') consists of:

$$f \quad : \quad \mathrm{Hom}_\Gamma(x, y) \tag{1.4.53}$$
$$g \quad : \quad \mathrm{Hom}_{A(y)}(A(f)(x), y) \tag{1.4.54}$$

Example 1.4.15. There is a variant of the previous example with groupoids instead of categories, using the groupoid of groupoids.

Example 1.4.16. The initial category with families is often used to represent the syntax of type theory. This gives a soundness test for a candidate syntax: its syntactic category should be equivalent to the initial category with families.

Remark 1.4.17. There exist many approaches to building a category with families having strict substitutions:

- Such a model can be built directly from a universe [**KL12a**], i.e. a well-behaved map:

$$p \quad : \quad \widetilde{U} \to U \tag{1.4.55}$$

  with chosen pullbacks along any map to U. A similar approach uses the so-called *local universes* [**LW15**].
- Alternatively, it is possible to transform a lex category into a model with strict substitutions [**Hof94**]. Similarly there is a biequivalence between locally cartesian closed categories and democratic categories with families with product, arrow and extensional identity types [**CD14**].

Remark 1.4.18. There exist many other notions of model of type theory, for example comprehension categories [**Jac93**], categories with attributes [**Car86**], display categories [**Tay99**], natural models [**Awo18**] and contextual categories [**Car86**]. We expect parametricity to work well with them, but this is out of scope for this thesis.

The definition for parametric category with families can be found in Section 2.6. The core idea is that a parametric category with families is a category with families $\mathcal{C}$ equipped with:

- For any context $\Gamma : \mathcal{C}$ a type:

$$\Gamma, \Gamma \quad \vdash \quad \Gamma_* \tag{1.4.56}$$

  This requires a product of context to be defined.

- For any substitution:

$$\sigma \quad : \quad \mathrm{Hom}_{\mathcal{C}}(\Gamma, \Delta) \tag{1.4.57}$$

a term:

$$\sigma_* \quad : \quad \mathrm{Tm}((\Gamma, \Gamma, \Gamma_*), \Delta_*[\sigma, \sigma]) \tag{1.4.58}$$

and a similar structure for types and terms. Moreover, this structure should obey many equations. For example, we should have:

$$(A \times B)_* \quad = \quad A_* \times B_* \tag{1.4.59}$$

$$\top_* \quad = \quad \top \tag{1.4.60}$$

We refer to Section 2.6 for the full list of equations. Building models obeying them is challenging, and we will present a method to do so in Chapter 2.

1.5. Variants of parametricity

In this section we survey the many variants of parametricity that have been considered in the past. The parametricity where any type comes with a relation is called *standard*, as seen in the previous sections. We attempt to suggest a manifold of examples, rather than to give a full classification. It should be noted that there is no established consensus on the vocabulary presented here.

- ***n*-ary parametricity.** The relations assumed in the standard parametricity are binary. It is straightforward to give a variant called n-ary parametricity, where any type comes with an n-ary relation.

  Unary parametricity is often considered (see e.g. [**BCM15**]), although the binary version is more frequent in a homotopical context, as a path links *two* points. To my knowledge, there is no reference to n-ary parametricity for $n > 2$.

Remark 1.5.1. The notion of unary parametricity is similar to Kleene-style realisability, in the sense that any type comes with a predicate. But they have significant differences:

- In realisability, a formula A is sent to a predicate A_* on some programs, for example on λ-terms or Gödel numbers for recursive functions. Then a program p such that $A_*(p)$ holds is called a realiser for A. Any proof of A gives a realiser for A, but some realisers do not come from proofs. So we can have realisers for an unprovable A.
- In unary parametricity a type A is sent to a predicate A_* on the type A itself, and a term $a : A$ is send to $a_* : A_*(a)$. Here the realiser is a itself, as proved by a_*, so that type theory is its own language of realisers. Since A_* is a predicate over A, we cannot consider a realiser for a type without inhabitant.

Overall, parametricity and realisability have distinct goals:

- Realisability validates new formulas using computational justifications, i.e. it shows formulas consistent by finding programs realising them.
- Unary parametricity emphasizes the fact that terms are similar to programs, i.e. that they are continuous in some sense: they preserve the relevant predicates.

Remark 1.5.2. We can consider the 0-ary case, where Γ_* does not depend on Γ. In this case being parametric just means having an endomorphism $\_{}_*$.

- **Iterated / truncated parametricity.** With the standard parametricity, any type A comes with a new type A_* so that we also have A_{**} and so on. A variant of parametricity which can only be iterated a fixed number of times n is called n-truncated. They come in two flavors:
  - **Heterogeneous.** Here parametricity takes a type in some language to build a relation in another language. For example, in [**Rey83**] a relation is built in set theory from any type in system F. This prevents iterating.
    In this case a parametric model will actually be a pair of models with parametricity going from one to the other. We do not consider any heterogeneous variant in this thesis, since we are interested in unfolding iterations.
  - **Homogeneous.** Here parametricity can be iterated but it stops giving meaningful information after a while. For example 1-truncated parametricity for categories with families can be defined as standard parametricity plus the equation:

$$\Gamma_{**} \quad = \quad \top \tag{1.5.1}$$

- **External / internal parametricity.** The standard parametricity is called external because it cannot be used when reasoning inside a model. Indeed for:

$$\Gamma \quad \vdash \quad t : A \tag{1.5.2}$$

we have:

$$\Gamma_0, \Gamma_1, \Gamma_* \quad \vdash \quad t_* : A_*[t_0, t_1] \tag{1.5.3}$$

This rule changes the context so that it cannot be used internally.

Internally parametric type theories have been considered many times (e.g. [**BCM15**] and [**CH20**]), using different techniques. The most direct approach is to add reflexivities, more precisely:

  - For any context Γ we assume a term:

$$x : \Gamma \quad \vdash \quad \mathrm{refl}_\Gamma : \Gamma_*[x, x] \tag{1.5.4}$$

  - For any type:

$$\Gamma \quad \vdash \quad A \tag{1.5.5}$$

  we assume a term:

$$x : \Gamma, y : A \quad \vdash \quad \mathrm{refl}_A : A_*[x, x, \mathrm{refl}_\Gamma, y, y] \tag{1.5.6}$$

  - For any term:

$$\Gamma \quad \vdash \quad t : A \tag{1.5.7}$$

  we assume the equation:

$$x : \Gamma \quad \vdash \quad \mathrm{refl}_A[x, t] = t_*[x, x, \mathrm{refl}_\Gamma] \tag{1.5.8}$$

  - Moreover we assume that reflexivities interact well with all the structure of our model.

Beware that reflexivities are not compatible with arrow types as we do not know how to define:

$$\mathrm{refl}_{A \to B} \tag{1.5.9}$$

from refl_A and refl_B. We have a similar issue with universes, where we do not know how to define:

$$\mathrm{refl}_A \tag{1.5.10}$$

for $A : \mathcal{U}$. In this thesis we will restrain to variants of parametricity where this issue does not arise, so that we will not consider reflexivities with arrow types or a universe.

- **Univalent / relative parametricity.** Hints toward a *cubical type theory without interval* are given in [**AK17**]. It is in fact a variant of parametricity where any type comes with an equivalence rather than a relation. This is done using the so-called coercions, meaning that given a type:

$$\Gamma \quad \vdash \quad A \tag{1.5.11}$$

we have a term:

$$x_0, x_1 : \Gamma, \Gamma_*[x_0, x_1], y : A[x_0] \vdash \mathrm{coe}_A : A[x_1] \tag{1.5.12}$$

allowing to model transport in dependent types. This, with other similar assumptions, makes the relation A_* behaves as an identity type for A.

Remark 1.5.3. A form of parametricity with isomorphisms rather than relations was considered as early as [**Rob94**].

A variant of parametricity where we have such coercions will be called *univalent*. In [**CH20**] the axiom:

$$\mathcal{U}_*(A, B) \quad \simeq \quad A \to B \to \mathcal{U} \tag{1.5.13}$$

for internal parametricity is called the *relativity axiom*, so non-univalent variants of parametricity will be called *relative*.

At the moment there is no consensus on how univalent parametricity is best formalised. It does not fit in our framework, as it is internal with arrow types and a universe.

Remark 1.5.4. *Higher observational type theory* is an unpublished type theory with a form of internal univalent parametricity proposed by Altenkirch, Kaposi and Shulman. In this type theory, from a term:

$$x_1 : A_1, \cdots, x_n : A_n \quad \vdash \quad t : B \tag{1.5.14}$$

and k a natural number such that $k \leq n$, we get a new term in the context obtained from:

$$x_1 : A_1, \cdots, x_n : A_n \tag{1.5.15}$$

by keeping the first k variables identical, and replacing the last $n - k$ variables by two related variables. For $k = 0$ we get t_* and for $k = n$ we get $\mathrm{refl}_B[t]$. In practice this means that $\mathrm{refl}_B[t]$ is computed inductively on t.

We consider a few examples of notions of parametricity, from the point of view developed in this section:

- The standard parametricity is binary, external, iterated and relative.
- The original parametricity by Reynolds [**Rey83**] is binary, external, truncated heterogeneous (as it goes from system F to set theory) and relative.
- The model considered in [**AGJ14**] should obey a form of binary, internal, 1-truncated and relative parametricity, although this is not laid down in the article.
- The parametricity considered in [**BCM15**] is unary, internal, iterated and relative.
- The cubical type theory without interval hinted in [**AK17**] would enjoy a binary, internal, iterated and univalent form of parametricity.
- The model in [**GFO16**] is meant to obey a form of binary, external, 2-truncated and relative parametricity.
- The models considered in [**JS17**] are meant to enjoy binary, internal and relative forms of parametricity. The framework presented there gives n-truncated variants for any n, plus an iterated one.
- In [**TTS21**] a form of *univalent parametricity* is introduced. It is binary, internal, iterated and (of course) univalent. It assumes a univalent universe to begin with, so it cannot be used to justify univalence.
- Setoid type theory [**ABKT19**] satisfies a form of parametricity which is binary, internal, 1-truncated (as setoids are 1-dimensional) and univalent (as setoids have *transitive* and *symmetric* relations).
- In [**CH20**] a *bi-parametricity* is introduced: any type comes with two relations, called its bridge type and path type. Both are binary, internal and iterated, but the bridge relation is relative and the identity relation is univalent.

CHAPTER 2

Notions of parametricity as extensions by section

In [**Rey83**] or [**BJP10**], parametricity is proven inductively in the initial model. In this chapter, we will axiomatize this situation, and give abstract methods building parametric models. We will try to be as modular as possible in the notion of model of type theory used, by simply assuming that the category of models of type theory is the category of algebras for a signature for Quotient Inductive-Inductive Type (QIITs) [**KKA19**]. This should mean that it can be presented by any of the following:

- **An essentially algebraic theory** [**AR94**]**.** Such a theory is a multi-sorted algebraic theory with partial operations, with their domains defined by equations.
- **A generalized algebraic theory** [**Car86**]**.** Here operations are total but sorts can depend on each other.
- **A signature for QIITs** [**KKA19**]**.** This notion is a recent type-theoretic recasting of generalized algebraic theories. It puts an emphasis on initial algebras and their induction principles. We will use this notion because we want to axiomatize the fact that parametricity is inductively defined.
- **A lex category.** An algebra for a lex category $\mathcal{C}$ is simply a lex functor from $\mathcal{C}$ to sets. This encoding is useful when dealing with abstract locally presentable categories, but not very helpful when building concrete examples.

All these notions are expected to be equivalent, although to my knowledge there is no reference in the litterature.

In this chapter we prove that several functors forgetting parametricity have right adjoints, i.e. that some cofreely parametric models exist. To do this we define extensions by section, which are special extensions of signatures for QIITs, and prove that their associated forgetful functors have right adjoints. The intuition behind extensions by section is that they add inductively defined unary operations to a signature. The precise definition makes heavy use of [**KKA19**] and [**KK20**].

The chapter is organised as follows:

- In Section 2.1 we will introduce categorical extensions by section, which are forgetful functors of a particular form, and prove that they have right adjoints. They will serve as a tool to prove that forgetful functors coming from extension by section have right adjoint.
- In Section 2.2 we give a quick overview of QIITs signatures, and use them to define extensions by section. We give a few basic examples.
- In Section 2.3 we prove that for T' an extension by section of T we have that the forgetful functor:

$$U \quad : \quad \mathrm{Alg}_{T'} \to \mathrm{Alg}_T \tag{2.0.1}$$

has a right adjoint:

$$R \quad : \quad \mathrm{Alg}_T \to \mathrm{Alg}_{T'} \tag{2.0.2}$$

We do this by proving that such U are categorical extensions by section. We give a few basic examples of such right adjoints R.

- In Sections 2.4 to 2.7, we will prove that standard parametricity for categories, clans and category with families (with or without arrow types and a universe) are extensions by section.
- In Section 2.8 we will sketch setoid type theory [**ABKT19**] and univalent parametricity [**TTS21**] as extensions by section. We leave the full proofs for future work.

2.1. Categorical extension by section

The definition of categorical extension by section is inspired by Definition 1.1.3 where a category $\mathcal{C}$ was observed to be parametric if we had a section of the functor:

$$\mathrm{ev}_V \quad : \quad \mathcal{C}^G \to \mathcal{C} \tag{2.1.1}$$

sending a graph in $\mathcal{C}$ to its object of vertices.

Definition 2.1.1. A copointed endofunctor on a category $\mathcal{V}$ is an endofunctor:

$$E \quad : \quad \mathcal{V} \to \mathcal{V} \tag{2.1.2}$$

with a natural transformation:

$$d \quad : \quad E \to \mathrm{Id} \tag{2.1.3}$$

Definition 2.1.2. Assume given a copointed endofunctor (E, d) on $\mathcal{V}$. We give the following definitions:

- An (E, d)-coalgebra is an object $\mathcal{C}$ in $\mathcal{V}$ with a section s of:

$$d_{\mathcal{C}} \quad : \quad E(\mathcal{C}) \to \mathcal{C} \tag{2.1.4}$$

- A morphism of coalgebra from $(\mathcal{C}, s)$ to $(\mathcal{D}, e)$ consists of:

$$F \quad : \quad \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{D}) \tag{2.1.5}$$

  such that the following square commutes:

$$\begin{tikzcd} E(\mathcal{C}) \arrow[r, "E(F)"] & E(\mathcal{D}) \\ \mathcal{C} \arrow[u, "s"] \arrow[r, "F"'] & \mathcal{D} \arrow[u, "e"'] \end{tikzcd}$$

The category of coalgebras is denoted by:

$$\mathrm{CoAlg}_{\mathcal{V}}(E, d) \tag{2.1.6}$$

Remark 2.1.3. Intuitively:

- The category $\mathcal{V}$ is the category of models of type theory.
- For $\mathcal{C} : \mathcal{V}$ a model, a section of:

$$d_{\mathcal{C}} \quad : \quad E(\mathcal{C}) \to \mathcal{C} \tag{2.1.7}$$

  means that $\mathcal{C}$ is parametric.

Definition 2.1.4. A categorical extension by section is a forgetful functor of the form:

$$U \quad : \quad \mathrm{CoAlg}_{\mathcal{V}}(E, d) \to \mathcal{V} \tag{2.1.8}$$

for a copointed endofuntor (E, d) where $\mathcal{V}$ has small limits and E commutes with them.

Example 2.1.5. Assume $\mathcal{V}$ a symmetric monoidal closed category with small limits and:

$$\mathcal{G} \quad : \quad \mathcal{V} \tag{2.1.9}$$

$$V \quad : \quad \mathrm{Hom}_{\mathcal{V}}(1, \mathcal{G}) \tag{2.1.10}$$

Then we have a copointed endofunctor (E, d) defined by:

$$E(\mathcal{C}) \quad = \quad \mathcal{G} \multimap \mathcal{C} \tag{2.1.11}$$

$$d_{\mathcal{C}} \quad = \quad \mathrm{ev}_V : (\mathcal{G} \multimap \mathcal{C}) \to \mathcal{C} \tag{2.1.12}$$

giving a categorical extension by section.

Next theorem is one of the main result from this chapter. In categorical language, it means that cofree coalgebras exist for any copointed endofunctor commuting with limits. Its version for algebras over an unpointed endofunctor is very well-known (see for example [**Adá74**]). A generalisation of (the dual of) our result is presented in [**Kel80**], with the endofunctor only assumed to be accessible.

THEOREM 2.1.6. *Any categorical extension by section:*

$$U \quad : \quad \mathrm{CoAlg}_{\mathcal{V}}(E, d) \to \mathcal{V} \tag{2.1.13}$$

has a right adjoint.

PROOF. We define the right adjoint directly. First we define its action on objects and morphisms:

- We denote by $\triangle$ the category freely generated by an object, a functor and a natural transformation:

$$0 \quad : \quad \mathrm{Ob}_{\triangle} \tag{2.1.14}$$

$$E \quad : \quad \triangle \to \triangle \tag{2.1.15}$$

$$d \quad : \quad E \to \mathrm{Id} \tag{2.1.16}$$

- Assume given $\mathcal{D}$ in $\mathcal{V}$, we denote by F the unique functor from $\triangle$ to $\mathcal{V}$ such that:

$$F(0) \quad = \quad \mathcal{D} \tag{2.1.17}$$

$$F(E(n)) \quad = \quad E(F(n)) \tag{2.1.18}$$

$$F(d_n) \quad = \quad d_{F(n)} \tag{2.1.19}$$

- We define:

$$R(\mathcal{D}) \quad = \quad \lim_{n:\triangle} F(n) \tag{2.1.20}$$

Now we want a section of $d_{R(\mathcal{D})}$. Since E commutes with limits and F commutes with E and d, we have a commutative square:

$$\begin{tikzcd}
E(\lim_{n:\triangle} F(n)) \arrow[r, "d"] \arrow[d, "\cong"'] & \lim_{n:\triangle} F(n) \arrow[d, "\mathrm{id}"] \\
\lim_{n:\triangle} F(E(n)) \arrow[r, "\lim_{n:\triangle} F(d_n)"'] & \lim_{n:\triangle} F(n)
\end{tikzcd}$$

so it is enough to find a section of $\lim_{n:\triangle} F(d_n)$.

- We denote the projections by:

$$\pi_n \quad : \quad \lim_{n:\triangle} F(n) \to F(n) \tag{2.1.21}$$

Then the map:

$$(\pi_{E(n)})_{n:\triangle} \quad : \quad \lim_{n:\triangle} F(n) \to \lim_{n:\triangle} F(E(n)) \tag{2.1.22}$$

is well-defined, indeed for all:

$$\sigma \quad : \quad \mathrm{Hom}_{\triangle}(m, n) \tag{2.1.23}$$

we have a commutative triangle:

$$\begin{tikzcd}
 & \lim_{n:\triangle} F(n) \arrow[dl, "\pi_{E(m)}"'] \arrow[dr, "\pi_{E(n)}"] & \\
F(E(m)) \arrow[rr, "F(E(\sigma))"'] & & F(E(n))
\end{tikzcd}$$

by the projection rule applied to $E(\sigma)$.

- We check that it is a section. For all $n : \triangle$ we have that:

$$\pi_n \circ \lim_{n:\triangle} F(d_n) \circ (\pi_{E(n)})_{n:\triangle} \quad = \quad F(d_n) \circ \pi_n \circ (\pi_{E(n)})_{n:\triangle} \tag{2.1.24}$$

$$= \quad F(d_n) \circ \pi_{E(n)} \tag{2.1.25}$$

$$= \quad \pi_n \tag{2.1.26}$$

so that:

$$\lim_{n:\triangle} F(d_n) \circ (\pi_{E(n)})_{n:\triangle} \quad = \quad \mathrm{id} \tag{2.1.27}$$

So we have a functor:

$$R \quad : \quad \mathcal{V} \to \mathrm{CoAlg}_{\mathcal{V}}(E, d) \tag{2.1.28}$$

Defined by:

$$R(\mathcal{D}) \quad = \quad \lim_{n:\triangle} F(n) \tag{2.1.29}$$

with a section isomorphic to $(\pi_{E(n)})_{n:\triangle}$.

Now we prove that it is right adjoint to the forgetful functor. Assume given:

$$(\mathcal{C}, s) \quad : \quad \mathrm{CoAlg}_{\mathcal{V}}(E, d) \tag{2.1.30}$$

$$\mathcal{D} \quad : \quad \mathcal{V} \tag{2.1.31}$$

we need to prove that the following are naturally isomorphic:

- The set of maps:

$$G \quad : \quad \mathcal{C} \to \mathcal{D} \tag{2.1.32}$$

- The set of maps:

$$G \quad : \quad \mathcal{C} \to \lim_{n:\triangle} F(n) \tag{2.1.33}$$

such that we have:

$$\begin{tikzcd}
\mathcal{C} \arrow[r, "G"] \arrow[d, "s"'] & \lim_{n:\triangle} F(n) \arrow[d] \\
E(\mathcal{C}) \arrow[r, "E(G)"'] & E(\lim_{n:\triangle} F(n))
\end{tikzcd}$$

where the vertical map on the right is isomorphic to $(\pi_{E(n)})_{n:\triangle}$.

By the universal propriety of limits, the second item is naturally isomorphic to:

- The set of families of maps:

$$G_n \quad : \quad \mathcal{C} \to F(n) \tag{2.1.34}$$

for $n : \triangle$ such that for all:

$$\sigma \quad : \quad \mathrm{Hom}_{\triangle}(m, n) \tag{2.1.35}$$

we have:

$$F(\sigma) \circ G_m \quad = \quad G_n \tag{2.1.36}$$

and for all $n : \triangle$ we have:

$$G_{E(n)} \quad = \quad E(G_n) \circ s \tag{2.1.37}$$

It is clear that:

$$G_0 \quad : \quad \mathcal{C} \to \mathcal{D} \tag{2.1.38}$$

uniquely determines G_n for any n via the inductive definition:

$$G_{E(n)} \quad = \quad E(G_n) \circ s \tag{2.1.39}$$

To conclude we just need to prove that any G_0 determines a compatible family, i.e. we need to check that the family defined from G_0 by:

$$G_{E(n)} \quad = \quad E(G_n) \circ s \tag{2.1.40}$$

is such that for any:

$$\sigma \quad : \quad \mathrm{Hom}_{\triangle}(m, n) \tag{2.1.41}$$

we have:

$$F(\sigma) \circ G_m \quad = \quad G_n \tag{2.1.42}$$

This is done by induction on σ:

- If $\sigma = d_n$, then we have:

$$\begin{aligned}
F(d_n) \circ G_{E(n)} \quad &= \quad F(d_n) \circ E(G_n) \circ s && (2.1.43)\\
&= \quad d_{F(n)} \circ E(G_n) \circ s && (2.1.44)\\
&= \quad G_n \circ d_{\mathcal{C}} \circ s && (2.1.45)\\
&= \quad G_n && (2.1.46)
\end{aligned}$$

- We need to check that if the equation is true for σ, it is true for σ_*. Indeed:

$$\begin{aligned} F(E(\sigma)) \circ G_{E(m)} &= F(E(\sigma)) \circ E(G_m) \circ s && (2.1.47)\\ &= E(F(\sigma) \circ G_m) \circ s && (2.1.48)\\ &= E(G_n) \circ s && (2.1.49)\\ &= G_{E(n)} && (2.1.50)\end{aligned}$$

- Closure by composition and identity is immediate.

□

Remark 2.1.7. In fact $\triangle$ is the category of augmented semi-simplices. For any category $\mathcal{D}$, we can informally draw the functor:

$$F \quad : \quad \triangle \to \mathcal{D} \tag{2.1.51}$$

such that:

$$\begin{aligned} F(0) &= X && (2.1.52)\\ F(E(n)) &= E(F(n)) && (2.1.53)\\ F(d_n) &= d_{F(n)} && (2.1.54)\end{aligned}$$

as an augmented semi-simplicial diagram:

$$X \leftarrow E(X) \leftleftarrows E^2(X) \cdots \tag{2.1.55}$$

in $\mathcal{D}$.

Remark 2.1.8. Assume given a categorical extension by section as in Example 2.1.5. This means that $\mathcal{V}$ is a monoidal symmetric closed category and we have:

$$\begin{aligned} \mathcal{G} &: \mathrm{Ob}_{\mathcal{V}} && (2.1.56)\\ V &: \mathrm{Hom}_{\mathcal{V}}(1, \mathcal{G}) && (2.1.57)\end{aligned}$$

generating an extension by section via the copointed endofunctor (E, d) defined by:

$$\begin{aligned} E(\mathcal{C}) &= \mathcal{G} \multimap \mathcal{C} && (2.1.58)\\ d_{\mathcal{C}} &= \mathrm{ev}_V : (\mathcal{G} \multimap \mathcal{C}) \to \mathcal{C} && (2.1.59)\end{aligned}$$

Then the right adjoint R is such that:

$$\begin{aligned} R(\mathcal{C}) &\cong \lim\big((1 \multimap \mathcal{C}) \leftarrow (\mathcal{G} \multimap \mathcal{C}) \leftleftarrows ((\mathcal{G} \otimes \mathcal{G}) \multimap \mathcal{C}) \ \cdots\big) && (2.1.60)\\ &\cong \mathrm{colim}(1 \to \mathcal{G} \rightrightarrows \mathcal{G} \otimes \mathcal{G} \ \cdots) \multimap \mathcal{C} && (2.1.61)\end{aligned}$$

But the colimit on the left is the free monoid in $\mathcal{V}$ generated by $\mathcal{G}$ with unit V. In Chapter 3 we will extend this to any monoid in $\mathcal{V}$.

2.2. Quotient inductive-inductive types and extensions by section

In this section, we will sketch Quotient Inductive-Inductive Types (QIITs) following [**KKA19**] and [**KK20**], and then use them to define extensions by section. This section cannot be fully understood without some familiarity with these sources.

Remark 2.2.1. We clarify the vocabulary on quotient inductive-inductive types:

- A signature for QIITs (abbreviated as a signature) is a syntactic object, for example the theory of rings.

- An algebra for a signature consists of data obeying the rules specified by the signature, for example a ring.
- A quotient inductive-inductive type is an initial algebra for a signature, for example the ring $\mathbb{Z}$.

A signature is defined as a context in a type theory with:

- A universe.
- Arrow types with domain in the universe.
- Extensional identity types for types in the universe. The universe is assumed closed under them.
- Unit and product types. The universe is assumed closed under them.

This is a mix of the signatures from [**KKA19**] and [**KK20**]. Indeed in [**KKA19**] the universe is not assumed closed under anything, and in [**KK20**] the identity types are not extensional.

The fact that arrow types have domains in $\mathcal{U}$ and that $\mathcal{U}$ is not stable under them is crucial. Indeed it enforces the strict positivity of types. Identity types are isomorphic to the meta-theoretic equality, so they have at most one inhabitant.

Remark 2.2.2. In [**KKA19**] and [**KK20**], an arrow type with a meta-theoretic domain is assumed, allowing for infinitary constructors. We have no need for them here so we do not assume them, but we could do so without trouble.

Now we give examples of signature, using type-theoretic notations.

Example 2.2.3. The signature for natural numbers is given by:

$$X : \mathcal{U} \tag{2.2.1}$$
$$0 : X \tag{2.2.2}$$
$$s : X \to X \tag{2.2.3}$$

Example 2.2.4. The signature for semi-groups is given by:

$$X : \mathcal{U} \tag{2.2.4}$$
$$m : X \to X \to X \tag{2.2.5}$$
$$\_ : (x, y, z : X) \to \mathrm{Id}(m(m(x, y), z), m(x, m(y, z))) \tag{2.2.6}$$

Example 2.2.5. The signature for reflexive graphs is given by:

$$V : \mathcal{U} \tag{2.2.7}$$
$$E : V \to V \to \mathcal{U} \tag{2.2.8}$$
$$r : (x : V) \to E(x, x) \tag{2.2.9}$$

Given a signature, we can define its category of algebras inductively on the signature. It is explained precisely how to do this in:

- Section 4 of [**KKA19**] and Section 5 of [**KK20**] for objects.
- Section 5 of [**KKA19**] and Section 7 of [**KK20**] for morphisms.

These definitions of algebras and their morphisms depend on a target theory. For Γ a signature, we denote by Alg_Γ its category of algebras using sets as a target. We give an example:

Example 2.2.6. For Γ the signature of natural numbers, its category of algebras is as follows:

- An object in Alg_Γ is a set X with:

$$0_X \quad : \quad X \tag{2.2.10}$$
$$s_X \quad : \quad X \to X \tag{2.2.11}$$

- A morphism in Alg_Γ from $(X, 0_X, s_X)$ to $(Y, 0_Y, s_Y)$ consists of:

$$f \quad : \quad X \to Y \tag{2.2.12}$$

  such that:

$$f(0_X) \quad = \quad 0_Y \tag{2.2.13}$$

  and for all $x : X$ we have:

$$f(s_X(x)) \quad = \quad s_Y(f(x)) \tag{2.2.14}$$

An initial object in any category of algebras is called a QIIT. It satisfies an induction principle defined in three steps:

(1) Displayed algebras over an algebra X are defined inductively on a signature (Section 6 of [**KKA19**] and [**KK20**]). Displayed algebras over X are equivalent to morphisms of algebras with target X.
(2) Sections of a displayed algebra Y over X are defined inductively on a signature (Section 6 of [**KKA19**] and Section 8 of [**KK20**]). A section of a displayed algebra is equivalent to a section of the corresponding morphism to X, hence the name.
(3) The induction principle asserts that any displayed algebra over the initial algebra has a section.

When using sets as a target, any signature has a QIIT in its category of algebras. Assuming the same using a type theory as target was the main point of [**KKA19**] and [**KK20**].

Example 2.2.7. The initial algebra for the signature of natural numbers is the set of natural numbers. Its induction principle is well-known. The whole theory of inductive types is inspired by this example.

Example 2.2.8. Consider (X, m) a semi-group.

- A displayed algebra over (X, m) consists of:

$$\widetilde{X} \quad : \quad X \to \mathrm{Set} \tag{2.2.15}$$
$$\widetilde{m} \quad : \quad \widetilde{X}(x) \to \widetilde{X}(y) \to \widetilde{X}(m(x, y)) \tag{2.2.16}$$

  such that given:

$$\widetilde{x} \quad : \quad \widetilde{X}(x) \tag{2.2.17}$$
$$\widetilde{y} \quad : \quad \widetilde{X}(y) \tag{2.2.18}$$
$$\widetilde{z} \quad : \quad \widetilde{X}(z) \tag{2.2.19}$$

  we have that:

$$\widetilde{m}(\widetilde{m}(\widetilde{x}, \widetilde{y}), \widetilde{z}) \quad = \quad \widetilde{m}(\widetilde{x}, \widetilde{m}(\widetilde{y}, \widetilde{z})) \tag{2.2.20}$$

  This equation is well-typed because:

$$m(m(x, y), z) \quad = \quad m(x, m(y, z)) \tag{2.2.21}$$

- A section of a displayed algebra $(\widetilde{X}, \widetilde{m})$ consists of:

$$s \quad : \quad (x : X) \to \widetilde{X}(x) \tag{2.2.22}$$

  such that for all $x, y : X$ we have:

$$s(m(x, y)) \quad = \quad \widetilde{m}(s(x), s(y)) \tag{2.2.23}$$

  There is no condition corresponding to associativity, because there is at most one inhabitant in any identity type.

Any displayed semi-group over the initial semi-group has a section. This implies that the initial semi-group is empty.

When defining algebras, displayed algebras and their sections we can also use the theory of signatures as target theory. For algebras this gives the identity translation. For displayed algebra we get the following:

Proposition 2.2.9. *For any signature* Γ*, there is a type:*

$$\mathrm{Disp}_\Gamma \quad : \quad \mathrm{Ty}(\Gamma) \tag{2.2.24}$$

in the theory of signatures, with a commutative triangle of functors:

$$\begin{tikzcd}
\mathrm{Alg}_{\Gamma,\mathrm{Disp}_\Gamma} \arrow[rr, "\simeq"] \arrow[dr, "U"'] & & \mathrm{Alg}_\Gamma^{\to} \arrow[dl, "\mathrm{cod}"] \\
& \mathrm{Alg}_\Gamma &
\end{tikzcd}$$

where:

- $\mathrm{Alg}_\Gamma^{\to}$ *is the category of arrows in* Alg_Γ.
- cod *is the target functor.*
- U *is the forgetful functor.*

Here Disp_Γ denote the type of displayed algebras over Γ, defined using the theory of signatures as target theory. We can do the same for sections, getting the following:

Proposition 2.2.10. *For any signature* Γ*, there is a type:*

$$\mathrm{Sec}_\Gamma \quad : \quad \mathrm{Ty}(\Gamma, \mathrm{Disp}_\Gamma) \tag{2.2.25}$$

in the theory of signatures, with a commutative square of functors:

$$\begin{tikzcd}
\mathrm{Alg}_{\Gamma,\mathrm{Disp}_\Gamma,\mathrm{Sec}_\Gamma} \arrow[r, "\simeq"] \arrow[d] & \mathrm{Alg}_\Gamma^{s} \arrow[d] \\
\mathrm{Alg}_{\Gamma,\mathrm{Disp}_\Gamma} \arrow[r, "\simeq"'] & \mathrm{Alg}_\Gamma^{\to}
\end{tikzcd}$$

where:

- Alg_Γ^{s} *is the category of arrows with a section in* Alg_Γ.
- *The vertical arrows are forgetful functors.*

Here Sec_Γ denote the type of sections of a displayed algebras over Γ, defined using the theory of signatures as a target. We do not prove both previous propositions, which essentially assert that:

- Displayed algebras over Γ indeed correspond to morphisms to Γ.

- Sections of displayed algebras over Γ correspond to sections of the corresponding morphism to Γ.

As we consider algebras using sets as a target, these two propositions rely crucially on the fact that families of sets indexed by Γ are equivalent to maps to Γ.

Now we can define extensions by section. The main examples we have in mind are notions of parametricity, as we will see in Sections 2.4 to 2.7.

Definition 2.2.11. Assume given a signature Γ. Then an extension by section of Γ is an extension of the form:

$$\Gamma,\ \mathrm{Sec}_\Gamma[\mathrm{id}, a] \tag{2.2.26}$$

for some:

$$a \quad : \quad \mathrm{Tm}(\Gamma, \mathrm{Disp}_\Gamma) \tag{2.2.27}$$

in the theory of signature.

So an extension by section is an extension adding a section to a chosen displayed algebra. This displayed algebra needs to be defined in the theory of signature.

Remark 2.2.12. In [**Moe21**] we defined extensions by section as extensions by:

- Unary operations with equations defining them inductively.
- Inductively provable unary equations.

The definition we give is a more precise rephrasing of this.

Extensions by section are extensions of a very specific form, indeed any extension adding a constant or a binary operation cannot be an extension by section. We give a first example, with a detailed proof.

Example 2.2.13. The extension of:

$$X \quad : \quad \mathcal{U} \tag{2.2.28}$$

by:

$$s \quad : \quad X \to X \tag{2.2.29}$$

is an extension by section.

Proof. We have:

$$\mathrm{Disp}_{X:\mathcal{U}} \quad : \quad \mathrm{Ty}(X : \mathcal{U}) \tag{2.2.30}$$

$$\mathrm{Disp}_{X:\mathcal{U}} \quad = \quad X \to \mathcal{U} \tag{2.2.31}$$

and:

$$\mathrm{Sec}_{X:\mathcal{U}} \quad : \quad \mathrm{Ty}(X : \mathcal{U}, \widetilde{X} : X \to \mathcal{U}) \tag{2.2.32}$$

$$\mathrm{Sec}_{X:\mathcal{U}} \quad = \quad \Pi(x : X).\, \widetilde{X}(x) \tag{2.2.33}$$

Then we define:

$$a \quad : \quad \mathrm{Tm}(X : \mathcal{U}, X \to \mathcal{U}) \tag{2.2.34}$$

$$a \quad = \quad \lambda(\_ : X).\, X \tag{2.2.35}$$

And we have:

$$\mathrm{Sec}_{X:\mathcal{U}}[\mathrm{id}, a] \quad = \quad X \to X \tag{2.2.36}$$

So that:

$$X : \mathcal{U},\ X \to X \tag{2.2.37}$$

is indeed an extension by section of $X : \mathcal{U}$. $\square$

We started from $X : \mathcal{U}$ and we defined inductively a unary operation $s : X \to X$ by giving nothing. Here is another similar example:

Example 2.2.14. The extension of the theory of graphs:

$$\begin{aligned} V &: \mathcal{U} && (2.2.38)\\ E &: V \to V \to \mathcal{U} && (2.2.39)\end{aligned}$$

by reflexivities:

$$r \quad : \quad (x : V) \to E(x, x) \tag{2.2.40}$$

is an extension by section.

Proof. A displayed algebra over:

$$\begin{aligned} V &: \mathcal{U} && (2.2.41)\\ E &: V \to V \to \mathcal{U} && (2.2.42)\end{aligned}$$

consists of:

$$\begin{aligned} \widetilde{V} &: V \to \mathcal{U} && (2.2.43)\\ \widetilde{E} &: (x : V) \to \widetilde{V}(x) \to (y : V) \to \widetilde{V}(y) \to E(x, y) \to \mathcal{U} && (2.2.44)\end{aligned}$$

We define:

$$\begin{aligned} \widetilde{V} &= \lambda(x : V).\, E(x, x) && (2.2.45)\\ \widetilde{E} &= \lambda(\cdots).\, \top && (2.2.46)\end{aligned}$$

As there is no constructors for graphs, this an inductive definition with nothing to check. $\square$

Next example is more complicated:

Example 2.2.15. The theory of groups is an extension by section of the theory of monoids.

Proof. We proceed step-by-step:

- A monoid consists of:

$$\begin{aligned} X &: \mathcal{U} && (2.2.47)\\ m &: X \to X \to X && (2.2.48)\\ 1 &: X && (2.2.49)\end{aligned}$$

  such that:

$$\begin{aligned} m(1, x) &= x && (2.2.50)\\ m(x, 1) &= x && (2.2.51)\\ m(m(x, y), z) &= m(x, m(y, z)) && (2.2.52)\end{aligned}$$

  where x, y, z are implicitly universally quantified.

- A displayed algebra over a monoid $(X, m, 1)$ consists of:

$$\widetilde{X} : X \to \mathcal{U} \tag{2.2.53}$$
$$\widetilde{m} : \widetilde{X}(x) \to \widetilde{X}(y) \to \widetilde{X}(m(x,y)) \tag{2.2.54}$$
$$\widetilde{1} : \widetilde{X}(1) \tag{2.2.55}$$

 such that for any:

$$\widetilde{x} : \widetilde{X}(x) \tag{2.2.56}$$
$$\widetilde{y} : \widetilde{X}(y) \tag{2.2.57}$$
$$\widetilde{z} : \widetilde{X}(z) \tag{2.2.58}$$

 we have:

$$\widetilde{m}(\widetilde{1}, \widetilde{x}) = \widetilde{x} \tag{2.2.59}$$
$$\widetilde{m}(\widetilde{x}, \widetilde{1}) = \widetilde{x} \tag{2.2.60}$$
$$\widetilde{m}(\widetilde{m}(\widetilde{x}, \widetilde{y}), \widetilde{z}) = \widetilde{m}(\widetilde{x}, \widetilde{m}(\widetilde{y}, \widetilde{z})) \tag{2.2.61}$$

- A section of a displayed algebra $(\widetilde{X}, \widetilde{m}, \widetilde{1})$ consists of:

$$s : (x : X) \to \widetilde{X}(x) \tag{2.2.62}$$

 such that:

$$s(m(x,y)) = \widetilde{m}(s(x), s(y)) \tag{2.2.63}$$
$$s(1) = \widetilde{1} \tag{2.2.64}$$

- Now we define a displayed algebra by:

$$\widetilde{X}(x) = (x^{-1} : X) \times \mathrm{Id}(m(x, x^{-1}), 1) \times \mathrm{Id}(m(x^{-1}, x), 1) \tag{2.2.65}$$

 So $\widetilde{X}(x)$ holds when x is invertible. We need to prove that:
 - If x and y are invertible, so is $m(x, y)$. We prove this by defining:

$$(m(x,y))^{-1} = m(y^{-1}, x^{-1}) \tag{2.2.66}$$

 - The unit 1 is invertible. We prove this by defining:

$$1^{-1} = 1 \tag{2.2.67}$$

 The required equations on $\widetilde{m}$ and $\widetilde{1}$ are automatically true because $\widetilde{X}(x)$ is a proposition, i.e. it has at most one inhabitant.
- Finally a section of this displayed algebra consists of:

$$s : (x : X) \to (x^{-1} : X) \times \mathrm{Id}(m(x, x^{-1}), 1) \times \mathrm{Id}(m(x^{-1}, x), 1) \tag{2.2.68}$$

 such that:

$$(m(x,y))^{-1} = m(y^{-1}, x^{-1}) \tag{2.2.69}$$
$$1^{-1} = 1 \tag{2.2.70}$$

 But these equations are always true, so that s is precisely a witness that the monoid is a group.

□

The previous proof can be summarized by saying that inverses in a monoid can be inductively defined. From now on we will be less precise when constructing extensions by section, relying on the intuition that they add inductively defined unary operations.

2.3. Extensions by section have right adjoints

We want to check that an extension by section indeed induces a categorical extension by section, so that their associated forgetful functors have right adjoints. We give an auxiliary lemma:

Lemma 2.3.1. *Assume given a substitution:*

$$\sigma \quad : \quad \mathrm{Hom}(\Gamma, \Delta) \tag{2.3.1}$$

then the induced functor:

$$U \quad : \quad \mathrm{Alg}_\Gamma \to \mathrm{Alg}_\Delta \tag{2.3.2}$$

has a left adjoint.

Proof Sketch. The core idea is that the left adjoint:

$$L \quad : \quad \mathrm{Alg}_\Delta \to \mathrm{Alg}_\Gamma \tag{2.3.3}$$

sends $\delta : \mathrm{Alg}_\Delta$ to the Γ-algebra freely generated by:

$$\gamma \quad : \quad \mathrm{Alg}_\Gamma \tag{2.3.4}$$

$$\_ \quad : \quad \mathrm{Hom}_{\mathrm{Alg}_\Delta}(\delta, U(\gamma)) \tag{2.3.5}$$

This is mentioned (for essentially algebraic theories) in Section 15 of [**Car86**], as an extension of the similar result for algebraic theories [**Law63**]. □

Now we prove that extensions by section give categorical extensions by section.

Proposition 2.3.2. *Assume given Γ' an extension by section of Γ. Then the forgetful functor:*

$$U \quad : \quad \mathrm{Alg}_{\Gamma'} \to \mathrm{Alg}_\Gamma \tag{2.3.6}$$

is equivalent to a categorical extension by section.

Proof. It is well-known that Alg_Γ has small limits. As Γ' is an extension by section of Γ we have a term:

$$a \quad : \quad \mathrm{Tm}(\Gamma, \mathrm{Disp}_\Gamma) \tag{2.3.7}$$

such that:

$$\Gamma' \quad = \quad \Gamma, \mathrm{Sec}_\Gamma[\mathrm{id}, a] \tag{2.3.8}$$

- The substitution:

$$(\mathrm{id}, a) \quad : \quad \mathrm{Hom}(\Gamma, (\Gamma, \mathrm{Disp}_\Gamma)) \tag{2.3.9}$$

  induces a functor:

$$\alpha \quad : \quad \mathrm{Alg}_\Gamma \to \mathrm{Alg}_{\Gamma, \mathrm{Disp}_\Gamma} \tag{2.3.10}$$

  By Lemma 2.3.1 we have that α has a left adjoint, so that it commutes with limits.
- As we have:

$$\mathrm{w} \circ (\mathrm{id}, a) \quad = \quad \mathrm{id} \tag{2.3.11}$$

  the functor α is a section of the forgetful functor:

$$\mathrm{Alg}_{\Gamma, \mathrm{Disp}_\Gamma} \to \mathrm{Alg}_\Gamma \tag{2.3.12}$$

But by Proposition 2.2.9, this gives a section of the target functor:

$$\mathrm{cod} \quad : \quad \mathrm{Alg}_\Gamma^{\rightarrow} \to \mathrm{Alg}_\Gamma \tag{2.3.13}$$

This is precisely an endofunctor:

$$E \quad : \quad \mathrm{Alg}_\Gamma \to \mathrm{Alg}_\Gamma \tag{2.3.14}$$

with a natural transformation:

$$d \quad : \quad E \to \mathrm{Id} \tag{2.3.15}$$

The fact that α commutes with limits means that E commutes with limits.

- We defined a by the requirement that:

$$\Gamma' \quad = \quad \Gamma, \mathrm{Sec}_\Gamma[\mathrm{id}, a] \tag{2.3.16}$$

By Proposition 2.2.10 the type Sec_Γ encodes sections of displayed algebras, and the term a gives (E, d), so that $\mathrm{Alg}_{\Gamma'}$ is equivalent to:

$$\mathrm{CoAlg}_{\mathrm{Alg}_\Gamma}(E, d) \tag{2.3.17}$$

and we can conclude.

□

Now we know that extensions by section give rise to categorical extensions by section, and that categorical extensions by section have right adjoints. We are ready to conclude:

Theorem 2.3.3. *Assume given Γ' an extension by section of Γ. The forgetful functor:*

$$U \quad : \quad \mathrm{Alg}_{\Gamma'} \to \mathrm{Alg}_\Gamma \tag{2.3.18}$$

has a right adjoint.

Proof. An extension by section is a categorical extension by section using Proposition 2.3.2, and we conclude using Theorem 2.1.6. □

Remark 2.3.4. Theorem 2.1.6 gives an explicit description for the right adjoint using augmented semi-simplicial limits. This description is not very convenient when trying to analyze the right adjoint, in fact it is usually better to use the universal property directly.

Remark 2.3.5. An alternative proof of Theorem 2.3.3 is given in [**Moe21**], where extensions by section are called *interpretations*. In brief:

- The theory of locally finitely presentable categories is used to prove that the forgetful functor U coming from an interpretation has a right adjoint if and only if it commutes with finite colimits.
- It is proven by hand that U commutes with the initial objects and pushouts, using their definitions as QIITs.

Now we give some examples of right adjoint built from Theorem 2.3.3.

Example 2.3.6. We consider the theory of groups as an extension by section of the theory of monoids, as in Example 2.2.15. In this case the right adjoint is:

$$R \quad : \quad \mathrm{Mon} \to \mathrm{Grp} \tag{2.3.19}$$

$$R(M) \quad = \quad M^\times \tag{2.3.20}$$

where $M^\times$ is the group of invertible elements in M.

Indeed $\mathbb{Z}$ is the free group generated by $\{1\}$, so the underlying set of $R(M)$ is:

$$\begin{aligned} R(M) &\cong \mathrm{Hom}_{\mathrm{Set}}(\{1\}, R(M)) && (2.3.21)\\ &\cong \mathrm{Hom}_{\mathrm{Grp}}(\mathbb{Z}, R(M)) && (2.3.22)\\ &\cong \mathrm{Hom}_{\mathrm{Mon}}(\mathbb{Z}, M) && (2.3.23)\\ &\cong M^{\times} && (2.3.24)\end{aligned}$$

The group structure is computed the same way.

Example 2.3.7. We consider the theory of reflexive graphs as an extension by section of the theory of graphs, as in Example 2.2.14. Reflexive graphs are given by the theory:

$$\begin{aligned} V &: \mathrm{Set} && (2.3.25)\\ E &: V \to V \to \mathrm{Set} && (2.3.26)\\ r &: (x : V) \to E(x, x) && (2.3.27)\end{aligned}$$

The right adjoint:

$$R : \mathrm{Gph} \to \mathrm{rGph} \qquad (2.3.28)$$

sends a graph $G = (V, E)$ to the reflexive graph $R(G)$ defined by:

$$\begin{aligned} V_{R(G)} &= (x : V) \times E(x, x) && (2.3.29)\\ E_{R(G)} &= (x, e), (x', e') \mapsto E(x, x') && (2.3.30)\\ r_{R(G)} &= (x, e) \mapsto e && (2.3.31)\end{aligned}$$

To see this, we consider $\mathbb{I}_{\mathrm{v}}$ the reflexive graph freely generated by a vertex v. Then the set $V_{R(G)}$ of vertices of $R(G)$ is such that:

$$\begin{aligned} V_{R(G)} &\cong \mathrm{Hom}_{\mathrm{Set}}(\{\mathrm{v}\}, V_{R(G)}) && (2.3.32)\\ &\cong \mathrm{Hom}_{\mathrm{rGph}}(\mathbb{I}_{\mathrm{v}}, R(G)) && (2.3.33)\\ &\cong \mathrm{Hom}_{\mathrm{Gph}}(U(\mathbb{I}_{\mathrm{v}}), G) && (2.3.34)\\ &\cong (x : V) \times E(x, x) && (2.3.35)\end{aligned}$$

The rest of the structure is computed the same way.

Example 2.3.8. We consider the extension by section from Example 2.2.13, where:

$$X : \mathfrak{U} \qquad (2.3.36)$$

is extended by:

$$s : X \to X \qquad (2.3.37)$$

We denote its category of models by Set_s. Then the right adjoint:

$$R : \mathrm{Set} \to \mathrm{Set}_s \qquad (2.3.38)$$

sends a set X to:

$$R(X) = \mathbb{N} \to X \qquad (2.3.39)$$

with the function:

$$\begin{aligned} s &: (\mathbb{N} \to X) \to (\mathbb{N} \to X) && (2.3.40)\\ s(f) &= n \mapsto f(n + 1) && (2.3.41)\end{aligned}$$

To see this, we use the fact that $\mathbb{N}$ is the free object in Set_s generated by $\{0\}$. Then the underlying set of $R(X)$ is:

$$\begin{aligned} R(X) &\cong \mathrm{Hom}_{\mathrm{Set}}(\{0\}, R(X)) && (2.3.42)\\ &\cong \mathrm{Hom}_{\mathrm{Set}_s}(\mathbb{N}, R(X)) && (2.3.43)\\ &\cong \mathrm{Hom}_{\mathrm{Set}}(\mathbb{N}, X) && (2.3.44)\end{aligned}$$

The function:

$$f \mapsto (n \mapsto f(n+1)) \tag{2.3.45}$$

is computed the same way.

These three examples should be contrasted with each other:

- In the first example, being invertible is a property of an element in a monoid, because there is at most one inverse. Then the right adjoint sends a monoid to its group of invertible elements. This can be generalized to any inductively provable predicate, with the right adjoint sending an object to its subobject of elements obeying this predicate.
- In the second example, having an edge from x to x is really a structure on a vertex x, because there can be several such edges. So in this case we need to consider vertices together with a chosen edge in order to construct the right adjoint.
- The third example is the most interesting. Here having an image by s is a structure, so the right adjoint should consider any element together with its image by s. But this image should itself have an image by s, and so on. An iteration is taking place. This can be generalized to the extension by section of any theory by an endomorphism.

Now we are ready to introduce our main examples of extensions by section.

2.4. Parametricity for categories as an extension by section

Example 2.4.1. Categories are algebras for the following signature:

$$\begin{aligned} \mathrm{Ob} &: \mathcal{U} && (2.4.1)\\ \mathrm{Hom} &: \mathrm{Ob} \to \mathrm{Ob} \to \mathcal{U} && (2.4.2)\\ \mathrm{id} &: \mathrm{Hom}(\Gamma, \Gamma) && (2.4.3)\\ \_ \circ \_ &: \mathrm{Hom}(\Delta, \Theta) \to \mathrm{Hom}(\Gamma, \Delta) \to \mathrm{Hom}(\Gamma, \Theta) && (2.4.4)\end{aligned}$$

such that:

$$\begin{aligned} \mathrm{id} \circ \sigma &= \sigma && (2.4.5)\\ \sigma \circ \mathrm{id} &= \sigma && (2.4.6)\\ \sigma \circ (\delta \circ \theta) &= (\sigma \circ \delta) \circ \theta && (2.4.7)\end{aligned}$$

Example 2.4.2. Parametric categories are algebras for the signature for categories extended by:

$$\begin{aligned} \__* &: \mathrm{Ob} \to \mathrm{Ob} && (2.4.8)\\ \__* &: \mathrm{Hom}(\Gamma, \Delta) \to \mathrm{Hom}(\Gamma_*, \Delta_*) && (2.4.9)\\ d^0, d^1 &: \mathrm{Hom}(\Gamma_*, \Gamma) && (2.4.10)\end{aligned}$$

such that:

$$\begin{aligned} \mathrm{id}_* &= \mathrm{id} & (2.4.11)\\ (\sigma \circ \delta)_* &= \sigma_* \circ \delta_* & (2.4.12)\\ \sigma \circ d^0 &= d^0 \circ \sigma_* & (2.4.13)\\ \sigma \circ d^1 &= d^1 \circ \sigma_* & (2.4.14) \end{aligned}$$

Proposition 2.4.3. *The theory of parametric categories is an extension by section of the theory of categories.*

Proof. We define a displayed algebra over a category:

- First we define:

$$\begin{aligned} \widetilde{\mathrm{Ob}}(\Gamma) &= (\Gamma_* : \mathrm{Ob}) \times & (2.4.15)\\ & \quad (d^0, d^1 : \mathrm{Hom}(\Gamma_*, \Gamma)) & (2.4.16)\\ \widetilde{\mathrm{Hom}}(\sigma, (\Gamma_*, d^0_\Gamma, d^1_\Gamma), (\Delta_*, d^0_\Delta, d^1_\Delta)) &= (\sigma_* : \mathrm{Hom}(\Gamma_*, \Delta_*)) \times & (2.4.17)\\ & \quad (\sigma \circ d^0_\Gamma = d^0_\Delta \circ \sigma_*) \times & (2.4.18)\\ & \quad (\sigma \circ d^1_\Gamma = d^1_\Delta \circ \sigma_*) & (2.4.19) \end{aligned}$$

- For the identity we define:

$$\mathrm{id}_* = \mathrm{id} \qquad (2.4.20)$$

  and we check:

$$\begin{aligned} \mathrm{id} \circ d^0 &= d^0 \circ \mathrm{id}_* & (2.4.21)\\ \mathrm{id} \circ d^1 &= d^1 \circ \mathrm{id}_* & (2.4.22) \end{aligned}$$

- For composition, we define:

$$(\sigma \circ \delta)_* = \sigma_* \circ \delta_* \qquad (2.4.23)$$

  and we check that if d^0, d^1 are natural with respect to σ and δ, then they are natural with respect to $\sigma \circ \delta$.
- We need to check that our definition for $\_{}_*$ is compatible with the equations defining categories. Indeed for units we have:

$$\begin{aligned} (\sigma \circ \mathrm{id})_* &= \sigma_* \circ \mathrm{id} & (2.4.24)\\ &= \sigma_* & (2.4.25) \end{aligned}$$

  and:

$$\begin{aligned} (\mathrm{id} \circ \sigma)_* &= \mathrm{id} \circ \sigma_* & (2.4.26)\\ &= \sigma_* & (2.4.27) \end{aligned}$$

  and for composition we have:

$$\begin{aligned} (\sigma \circ (\delta \circ \theta))_* &= \sigma_* \circ (\delta_* \circ \theta_*) & (2.4.28)\\ &= (\sigma_* \circ \delta_*) \circ \theta_* & (2.4.29)\\ &= ((\sigma \circ \delta) \circ \theta)_* & (2.4.30) \end{aligned}$$

Then asking for a section of this displayed algebra is precisely asking that:

- Any $\Gamma : \mathrm{Ob}$ comes with:

$$\begin{aligned} \Gamma_* &: \mathrm{Ob} & (2.4.31)\\ d^0, d^1 &: \mathrm{Hom}(\Gamma_*, \Gamma) & (2.4.32) \end{aligned}$$

- Any morphism $\sigma : \mathrm{Hom}(\Gamma, \Delta)$ comes with:

$$\sigma_* \quad : \quad \mathrm{Hom}(\Gamma_*, \Delta_*) \tag{2.4.33}$$

  such that:

$$\sigma \circ d^0 \quad = \quad d^0 \circ \sigma_* \tag{2.4.34}$$

$$\sigma \circ d^1 \quad = \quad d^1 \circ \sigma_* \tag{2.4.35}$$

- For identity and compositions, we have that:

$$\mathrm{id}_* \quad = \quad \mathrm{id} \tag{2.4.36}$$

$$(\sigma \circ \delta)_* \quad = \quad \sigma_* \circ \delta_* \tag{2.4.37}$$

This is precisely a parametric category.

□

The previous proof emphasis two elementary but important facts:

- The equations for functors define them inductively.
- A transformation is always natural with respect to identities, and if it is natural with respect to σ and δ, then it is natural with respect to $\sigma \circ \delta$.

Example 2.4.4. From Proposition 2.4.3 with Theorem 2.3.3 we get that the forgetful functor from parametric categories to categories has a right adjoint R.

We denote by $\mathbb{I}_0$ the parametric category freely generated by an object 0. We have that for $\mathcal{C}$ a category:

$$\mathrm{Ob}_{R(\mathcal{C})} \quad \cong \quad \mathrm{Hom}_{\mathrm{Set}}(\{0\}, R(\mathcal{C})) \tag{2.4.38}$$

$$\cong \quad \mathrm{Hom}_{p\mathrm{Cat}}(\mathbb{I}_0, R(\mathcal{C})) \tag{2.4.39}$$

$$\cong \quad \mathrm{Hom}_{\mathrm{Cat}}(U(\mathbb{I}_0), \mathcal{C}) \tag{2.4.40}$$

The objects in $\mathbb{I}_0$ are:

$$0, 0_*, 0_{**}, \cdots \tag{2.4.41}$$

So giving an object in $X : \mathrm{Ob}_{R(\mathcal{C})}$ means giving a sequence of objects:

$$(\Gamma, \Gamma_*, \Gamma_{**}, \cdots) \tag{2.4.42}$$

in $\mathcal{C}$, together with some morphisms. For example we will have four morphisms:

$$d^0_{\Gamma_*}, d^1_{\Gamma_*}, (d^0_\Gamma)_*, (d^1_\Gamma)_* \tag{2.4.43}$$

going from Γ_{**} to Γ_*. By the naturality of d^0 and d^1, these morphisms have to obey many equations. We do not go into more detail here, but we will see in Example 3.3.2 that the category $R(\mathcal{C})$ is actually the category of semi-cubical objects in $\mathcal{C}$.

2.5. Parametricity for clans as an extension by section

We present clans and parametric clans as algebras for a signature. In order to do this we have to commit to morphisms between clans preserving limits on the nose, because morphisms between algebras commutes with operations on the nose.

Example 2.5.1. Categories with a terminal object are algebras for the signature of categories (Example 2.4.1) extended by:

$$\top \quad : \quad \mathrm{Ob} \tag{2.5.1}$$

$$\epsilon_\Gamma \quad : \quad \mathrm{Hom}(\Gamma, \top) \tag{2.5.2}$$

such that for any:

$$\sigma \quad : \quad \mathrm{Hom}(\Gamma, \top) \tag{2.5.3}$$

we have:

$$\sigma \quad = \quad \epsilon_\Gamma \tag{2.5.4}$$

Example 2.5.2. Clans are algebras for the signature of categories with terminal objects (Example 2.5.1) extended by:

- First we assume a predicate for fibrations:

$$\mathrm{Fib} \quad : \quad \mathrm{Hom}(\Gamma, \Delta) \to \mathcal{U} \tag{2.5.5}$$

  such that for any $e, e' : \mathrm{Fib}(\sigma)$ we have:

$$e \quad = \quad e' \tag{2.5.6}$$

  so we never have to name an inhabitant of $\mathrm{Fib}(\sigma)$. To say that $\mathrm{Fib}(\sigma)$ is inhabited we simply say that σ is a fibration.
- Moreover we ask that:
  - If σ and δ are fibrations, so is $\sigma \circ \delta$.
  - For any object Γ, we have that ϵ_Γ is a fibration.
  - If σ is an isomorphism, then σ is a fibration.
- We assume pullbacks, i.e. given:

$$\sigma \quad : \quad \mathrm{Hom}(\Delta, \Gamma) \tag{2.5.7}$$

$$p \quad : \quad \mathrm{Hom}(A, \Gamma) \tag{2.5.8}$$

  with p a fibration, we assume:

$$A \underset{\Gamma}{\times} \Delta \quad : \quad \mathrm{Ob} \tag{2.5.9}$$

$$\pi_1 \quad : \quad \mathrm{Hom}(A \underset{\Gamma}{\times} \Delta, A) \tag{2.5.10}$$

$$\pi_2 \quad : \quad \mathrm{Hom}(A \underset{\Gamma}{\times} \Delta, \Delta) \tag{2.5.11}$$

  such that π_2 is a fibration and:

$$p \circ \pi_1 \quad = \quad \sigma \circ \pi_2 \tag{2.5.12}$$

  Moreover we ask for the universal propriety of pullbacks, i.e. given any:

$$\delta \quad : \quad \mathrm{Hom}(\Theta, A) \tag{2.5.13}$$

$$\theta \quad : \quad \mathrm{Hom}(\Theta, \Delta) \tag{2.5.14}$$

  such that:

$$p \circ \delta \quad = \quad \sigma \circ \theta \tag{2.5.15}$$

  we have:

$$(\delta, \theta) \quad : \quad \mathrm{Hom}(\Theta, A \underset{\Gamma}{\times} \Delta) \tag{2.5.16}$$

  such that:

$$\pi_1 \circ (\delta, \theta) \quad = \quad \delta \tag{2.5.17}$$

$$\pi_2 \circ (\delta, \theta) \quad = \quad \theta \tag{2.5.18}$$

  Finally for any:

$$\sigma \quad : \quad \mathrm{Hom}(\Theta, A \underset{\Gamma}{\times} \Delta) \tag{2.5.19}$$

we ask that:

$$(\pi_1 \circ \sigma, \pi_2 \circ \sigma) \quad = \quad \sigma \tag{2.5.20}$$

In the previous definition we asked for isomorphisms to be fibrations, whereas in Definition 1.2.3 we assumed that fibrations were stable under isomorphisms. Both are equivalent because fibrations form a subcategory.

From now on we will write:

$$\sigma \quad : \quad \Gamma \longrightarrow \Delta \tag{2.5.21}$$

for σ an inhabitant of $\mathrm{Hom}(\Gamma, \Delta)$, and write:

$$p \quad : \quad A \twoheadrightarrow \Gamma \tag{2.5.22}$$

for p a fibration in $\mathrm{Hom}(A, \Gamma)$.

Example 2.5.3. Parametric clans are algebras for the signature for clans extended by parametricity for its underlying category, such that:

- The endofunctor $\_{}_*$ is a morphism of clans, meaning that if σ is a fibration, so is σ_*, and we have:

$$\begin{aligned}
\top_* &= \top && (2.5.23)\\
(\epsilon_\Gamma)_* &= \epsilon_{\Gamma_*} && (2.5.24)\\
(A \underset{\Gamma}{\times} \Delta)_* &= A_* \underset{\Gamma_*}{\times} \Delta_* && (2.5.25)\\
(\pi_1)_* &= \pi_1 && (2.5.26)\\
(\pi_2)_* &= \pi_2 && (2.5.27)\\
(\delta, \theta)_* &= (\delta_*, \theta_*) && (2.5.28)
\end{aligned}$$

- Moreover the condition on fibrations can be expressed by saying that given a fibration:

$$p \quad : \quad A \twoheadrightarrow \Gamma \tag{2.5.29}$$

we have that:

$$((d^0_A, d^1_A), p_*) \quad : \quad A_* \longrightarrow (A \times A) \underset{\Gamma \times \Gamma}{\times} \Gamma_* \tag{2.5.30}$$

is a fibration. This map is well defined because we have a commutative diagram:

$$\begin{tikzcd}
A_* \arrow[r, "p_*"] \arrow[d, "(d^0_A, d^1_A)"'] & \Gamma_* \arrow[d, "(d^0_\Gamma, d^1_\Gamma)"] \\
A \times A \arrow[r, "p \times p"'] & \Gamma \times \Gamma
\end{tikzcd}$$

Now we are ready for the expected claim:

Proposition 2.5.4. *The theory of parametric clans is an extension by section of the theory of clans.*

Proof. We define the parametricity structure inductively:

- **(Defining $\_*$).** Adding an endomorphism to a clan is indeed an extension by section, as is adding an endomorphism to anything. Indeed an endomorphism is inductively defined by the fact that it commutes with everything.

- **(Defining d^0, d^1).** It is enough to check that adding a natural transformation d from $\__*$ to the identity is an extension by section. We already did this for composition and identity, so we just do it for limits:
  - **(On $\top$).** We define:

$$d_\top \quad = \quad \mathrm{id}_\top \tag{2.5.31}$$

    This equation is valid in any clan, as both morphisms have target $\top$.
  - **(On ϵ_Γ).** Now need to prove that d is natural with respect to ϵ_Γ, i.e. that:

$$d_\top \circ (\epsilon_\Gamma)_* \quad = \quad \epsilon_\Gamma \circ d_\Gamma \tag{2.5.32}$$

    but both maps have target $\top$.
  - **(On $A \underset{\Gamma}{\times} \Delta$).** Assuming:

$$p \quad : \quad A \twoheadrightarrow \Gamma \tag{2.5.33}$$
$$\sigma \quad : \quad \Delta \to \Gamma \tag{2.5.34}$$

    we define:

$$d_{A \underset{\Gamma}{\times} \Delta} \quad = \quad (d_A \circ \pi_1, d_\Delta \circ \pi_2) \tag{2.5.35}$$

    This equation is true in any clan by the naturality of d relative to π_1 and π_2. The right-hand side is well-defined because:

$$\begin{aligned} p \circ d_A \circ \pi_1 &= d_\Gamma \circ p_* \circ \pi_1 && (2.5.36)\\ &= d_\Gamma \circ (p \circ \pi_1)_* && (2.5.37)\\ &= d_\Gamma \circ (\sigma \circ \pi_2)_* && (2.5.38)\\ &= d_\Gamma \circ \sigma_* \circ \pi_2 && (2.5.39)\\ &= \sigma \circ d_\Delta \circ \pi_2 && (2.5.40)\end{aligned}$$

  - **(On π_1 and π_2).** We need to prove that d is natural with respect to π_1 (and π_2 is similar):

$$\begin{aligned} d_A \circ (\pi_1)_* &= d_A \circ \pi_1 && (2.5.41)\\ &= \pi_1 \circ (d_A \circ \pi_1, d_\Delta \circ \pi_2) && (2.5.42)\\ &= \pi_1 \circ d_{A \underset{\Gamma}{\times} \Delta} && (2.5.43)\end{aligned}$$

  - **(On (p, q)).** Assuming that d is natural with respect to:

$$\delta \quad : \quad \Theta \to A \tag{2.5.44}$$
$$\theta \quad : \quad \Theta \to \Delta \tag{2.5.45}$$

    such that:

$$p \circ \delta \quad = \quad \sigma \circ \theta \tag{2.5.46}$$

    we need to prove d natural with respect to:

$$(\delta, \theta) \quad : \quad \Theta \to A \underset{\Gamma}{\times} \Delta \tag{2.5.47}$$

Indeed:

$$\begin{aligned}
d_{A \underset{\Gamma}{\times} \Delta} \circ (\delta, \theta)_* &= (d_A \circ \pi_1, d_\Delta \circ \pi_2) \circ (\delta_*, \theta_*) && (2.5.48)\\
&= (d_A \circ \delta_*, d_\Delta \circ \theta_*) && (2.5.49)\\
&= (\delta \circ d_\Theta, \theta \circ d_\Theta) && (2.5.50)\\
&= (\delta, \theta) \circ d_\Theta && (2.5.51)
\end{aligned}$$

where Equation 2.5.50 holds because d is natural with respect to δ and θ.

– **(Respecting equations)**. Now we need to prove that all the given inductive definitions respect the equations of clans. There are no equations between objects. From a morphism we defined proofs of naturality, but being natural is a proposition so equations between morphisms are automatically preserved.

- **(Proving the condition on fibrations)**. For any fibration:

$$p \quad : \quad A \twoheadrightarrow \Gamma \tag{2.5.52}$$

we denote by $\widehat{p}$ the map:

$$((d_A^0, d_A^1), p_*) \quad : \quad A_* \to (A \times A) \underset{\Gamma \times \Gamma}{\times} \Gamma_* \tag{2.5.53}$$

and for any object Γ we denote by $\widehat{\Gamma}$ the map:

$$(d_\Gamma^0, d_\Gamma^1) \quad : \quad \Gamma_* \to \Gamma \times \Gamma \tag{2.5.54}$$

We want to prove inductively on the fibration p that $\widehat{p}$ is a fibration. To make this induction go through we need to simultaneously prove that for any object Γ the map $\widehat{\Gamma}$ is a fibration.

– **(For $\top$)**. We prove that the map:

$$\widehat{\top} \quad : \quad \top_* \to \top \times \top \tag{2.5.55}$$

is a fibration. Indeed both objects are terminal so $\widehat{\top}$ is an isomorphism, therefore a fibration.

– **(For ϵ_Γ)**. We need to prove that:

$$\widehat{\epsilon_\Gamma} \quad : \quad \Gamma_* \to (\Gamma \times \Gamma) \underset{\top \times \top}{\times} \top_* \tag{2.5.56}$$

is a fibration, but this map is isomorphic to:

$$\widehat{\Gamma} \quad : \quad \Gamma_* \to \Gamma \times \Gamma \tag{2.5.57}$$

which is a fibration by the additional induction hypothesis on objects.

– **(For $A \underset{\Gamma}{\times} \Delta$)**. Assuming:

$$\begin{aligned}
p &: A \twoheadrightarrow \Gamma && (2.5.58)\\
\sigma &: \Delta \to \Gamma && (2.5.59)
\end{aligned}$$

we need to prove that:

$$\widehat{A \underset{\Gamma}{\times} \Delta} \quad : \quad (A \underset{\Gamma}{\times} \Delta)_* \to (A \underset{\Gamma}{\times} \Delta) \times (A \underset{\Gamma}{\times} \Delta) \tag{2.5.60}$$

is a fibration. But this map is isomorphic to the map:

$$\widehat{A} \underset{\widehat{\Gamma}}{\times} \widehat{\Delta} \quad : \quad A_* \underset{\Gamma_*}{\times} \Delta_* \to (A \times A) \underset{\Gamma \times \Gamma}{\times} (\Delta \times \Delta) \tag{2.5.61}$$

which is a fibration by the induction hypothesis on A, Γ and Δ.

– **(For $\pi_1$and π_2)**. We just do π_1. We need to prove that the induced map:

$$\widehat{\pi_1} \quad : \quad (A \underset{\Gamma}{\times} \Delta)_* \to (A \underset{\Gamma}{\times} \Delta) \times (A \underset{\Gamma}{\times} \Delta) \underset{A \times A}{\times} A_* \tag{2.5.62}$$

is a fibration. But this map is isomorphic to:

$$\mathrm{id}_{A_*} \underset{\widehat{\Gamma}}{\times} \widehat{\Delta} \quad : \quad A_* \underset{\Gamma_*}{\times} \Delta_* \to A_* \underset{\Gamma \times \Gamma}{\times} (\Delta \times \Delta) \tag{2.5.63}$$

which is a fibration by the induction hypothesis on Γ and Δ.

– **(For the composition)**. Assume given two fibrations:

$$p \quad : \quad A \twoheadrightarrow \Gamma \tag{2.5.64}$$

$$q \quad : \quad B \twoheadrightarrow A \tag{2.5.65}$$

such that we have induced fibrations:

$$\widehat{p} \quad : \quad A_* \twoheadrightarrow (A \times A) \underset{\Gamma \times \Gamma}{\times} \Gamma_* \tag{2.5.66}$$

$$\widehat{q} \quad : \quad B_* \twoheadrightarrow (B \times B) \underset{A \times A}{\times} A_* \tag{2.5.67}$$

We need to prove that:

$$\widehat{p \circ q} \quad : \quad B_* \to (B \times B) \underset{\Gamma \times \Gamma}{\times} \Gamma_* \tag{2.5.68}$$

is a fibration, but this map is isomorphic to:

$$((B \times B) \underset{A \times A}{\times} \widehat{p}) \circ \widehat{q} \quad : \quad B_* \to (B \times B) \underset{A \times A}{\times} (A \times A) \underset{\Gamma \times \Gamma}{\times} \Gamma_* \tag{2.5.69}$$

which is a fibration as the composition of fibrations.

– **(For the isomorphisms)**. For an isomorphism:

$$\sigma \quad : \quad \Gamma \to \Delta \tag{2.5.70}$$

we need to prove that the map:

$$\widehat{\sigma} \quad : \quad \Gamma_* \to (\Gamma \times \Gamma) \underset{\Delta \times \Delta}{\times} \Delta_* \tag{2.5.71}$$

is a fibration, but this map is isomorphic to the map:

$$\sigma_* \quad : \quad \Gamma_* \to \Delta_* \tag{2.5.72}$$

which is a fibration because σ is.

– **(Respecting equations)**. This respect any equation, as being a fibration is a proposition.

□

The two main points about the previous proof are that:

- The stability condition for fibration is provable by induction (if we extend it to objects).
- Natural transformations between morphisms of clan always commute with limits.

Remark 2.5.5. Technically we have to define $\_{}_*$, d^0 and d^1 and prove the condition on fibrations all in the same induction. This does not cause any issue here.

Example 2.5.6. We can build a right adjoint R to the forgetful functor from parametric clans to clans, using Proposition 2.5.4 and Theorem 2.3.3. It is very similar to the right adjoint for categories in Example 2.4.4, with the restriction that some morphisms should be fibrations.

We will see in Section 3.9 that the category $R(\mathcal{C})$ is in fact the clan of Reedy fibrant semi-cubical objects in $\mathcal{C}$.

2.6. Parametricity for categories with families as an extension by section

In this section we will see that categories with families and their parametric counterpart are algebras for a signature. Moreover we will prove that this gives an extension by section of the signature for categories with families.

For the sake of notational simplicity we consider unary parametricity, thus avoiding the need for products of contexts. As for clans, defining categories with families as algebras for a signature means considering strict morphisms between them, i.e. morphisms commuting with operations up to equality.

Definition 2.6.1. Categories with families are algebras for the signature for categories with a terminal object (Definition 2.5.1) extended by:

$$\mathrm{Ty} : \mathrm{Ob} \to \mathcal{U} \tag{2.6.1}$$
$$\mathrm{Tm} : (\Gamma : \mathrm{Ob}) \to \mathrm{Ty}(\Gamma) \to \mathcal{U} \tag{2.6.2}$$

with substitutions:

$$\_[\_] : \mathrm{Ty}(\Gamma) \to \mathrm{Hom}(\Delta, \Gamma) \to \mathrm{Ty}(\Delta) \tag{2.6.3}$$
$$\_[\_] : \mathrm{Tm}(\Gamma, A) \to (\sigma : \mathrm{Hom}(\Delta, \Gamma)) \to \mathrm{Tm}(\Delta, A[\sigma]) \tag{2.6.4}$$

such that for $A : \mathrm{Ty}(\Gamma)$ and $t : \mathrm{Tm}(\Gamma, A)$ we have:

$$A[\sigma \circ \delta] = A[\sigma][\delta] \tag{2.6.5}$$
$$A[\mathrm{id}] = A \tag{2.6.6}$$
$$t[\sigma \circ \delta] = t[\sigma][\delta] \tag{2.6.7}$$
$$t[\mathrm{id}] = t \tag{2.6.8}$$

with context comprehension:

$$(\_,\_) : (\Gamma : \mathrm{Ob}) \to \mathrm{Ty}(\Gamma) \to \mathrm{Ob} \tag{2.6.9}$$
$$(\_,\_) : (\sigma : \mathrm{Hom}(\Delta, \Gamma)) \to \mathrm{Tm}(\Delta, A[\sigma]) \to \mathrm{Hom}(\Delta, (\Gamma, A)) \tag{2.6.10}$$
$$\pi_1 : \mathrm{Hom}(\Delta, (\Gamma, A)) \to \mathrm{Hom}(\Delta, \Gamma) \tag{2.6.11}$$
$$\pi_2 : (\sigma : \mathrm{Hom}(\Delta, (\Gamma, A))) \to \mathrm{Tm}(\Delta, A[\pi_1(\sigma)]) \tag{2.6.12}$$
$$\tag{2.6.13}$$

such that:

$$\pi_1(\sigma, t) = \sigma \tag{2.6.14}$$
$$\pi_2(\sigma, t) = t \tag{2.6.15}$$
$$(\pi_1(\sigma), \pi_2(\sigma)) = \sigma \tag{2.6.16}$$
$$(\sigma, t) \circ \delta = (\sigma \circ \delta, t[\delta]) \tag{2.6.17}$$

In this definition, some equations require the previous ones to be well-typed. Categories with families axiomatise substitutions, so their theory is sometimes called the calculus of substitutions. Now we want to define product and unit types for categories with families.

We introduce some useful notations:

Notation 2.6.2. We define for $\Gamma : \mathrm{Ob}$ and $A : \mathrm{Ty}(\Gamma)$:

$$\mathrm{w} : \mathrm{Hom}((\Gamma, A), \Gamma) \tag{2.6.18}$$
$$\mathrm{w} = \pi_1(\mathrm{id}) \tag{2.6.19}$$

and:

$$\mathrm{v} : \mathrm{Tm}((\Gamma, A), A[\mathrm{w}]) \tag{2.6.20}$$
$$\mathrm{v} = \pi_2(\mathrm{id}) \tag{2.6.21}$$

Here w stands for *weakening* and v for *variable*. Then $\mathrm{v}[\mathrm{w}^n]$, where w^n is w composed n times, is similar to the de Bruijn index n.

Definition 2.6.3. Unit types for a category with families consist of:

$$\top : \mathrm{Ty}(\Gamma) \tag{2.6.22}$$
$$\epsilon : \mathrm{Tm}(\Gamma, \top) \tag{2.6.23}$$

such that for all:

$$t : \mathrm{Tm}(\Gamma, \top) \tag{2.6.24}$$

we have:

$$t = \epsilon \tag{2.6.25}$$

with equations for substitutions:

$$\top[\sigma] = \top \tag{2.6.26}$$
$$\epsilon[\sigma] = \epsilon \tag{2.6.27}$$

Definition 2.6.4. Product types for a category with families consist of:

$$\Sigma : (A : \mathrm{Ty}(\Gamma)) \to \mathrm{Ty}(\Gamma, A) \to \mathrm{Ty}(\Gamma) \tag{2.6.28}$$
$$(\_,\_) : (t : \mathrm{Tm}(\Gamma, A)) \to \mathrm{Tm}(\Gamma, B[\mathrm{id}, t]) \to \mathrm{Tm}(\Gamma, \Sigma(A, B)) \tag{2.6.29}$$
$$\pi_1 : \mathrm{Tm}(\Gamma, \Sigma(A, B)) \to \mathrm{Tm}(\Gamma, A) \tag{2.6.30}$$
$$\pi_2 : (t : \mathrm{Tm}(\Gamma, \Sigma(A, B)) \to \mathrm{Tm}(\Gamma, B[\mathrm{id}, \pi_1(t)] \tag{2.6.31}$$

such that we have:

$$\pi_1(s, t) = s \tag{2.6.32}$$
$$\pi_2(s, t) = t \tag{2.6.33}$$
$$(\pi_1(t), \pi_2(t)) = t \tag{2.6.34}$$

with equations for substitutions:

$$\Sigma(A, B)[\sigma] = \Sigma(A[\sigma], B[\sigma \circ \mathrm{w}, \mathrm{v}]) \tag{2.6.35}$$
$$(s, t)[\sigma] = (s[\sigma], t[\sigma]) \tag{2.6.36}$$

Remark 2.6.5. The equations for product types imply that:

$$\pi_1(t)[\sigma] = \pi_1(t[\sigma]) \tag{2.6.37}$$
$$\pi_2(t)[\sigma] = \pi_2(t[\sigma]) \tag{2.6.38}$$

Remark 2.6.6. The notation are overloaded:

- By $\top$ we can denote an object or a type, and by ϵ a morphism or a term..
- By $(\_,\_)$ we can denote a context, a substitution or a term, and similarly with π_1 and π_2.

Now we can define parametric categories with families. Recall that in this section we consider unary parametricity.

Definition 2.6.7. A parametric category with families is an algebra for the signature for category with families with unit and product types, extended by:

$$\_{}_* \;:\; (\Gamma : \mathrm{Ob}) \longrightarrow \mathrm{Ty}(\Gamma) \tag{2.6.39}$$
$$\_{}_* \;:\; (A : \mathrm{Ty}(\Gamma)) \longrightarrow \mathrm{Ty}(\Gamma, \Gamma_*, A[\mathrm{w}]) \tag{2.6.40}$$
$$\_{}_* \;:\; (\sigma : \mathrm{Hom}(\Gamma, \Delta)) \longrightarrow \mathrm{Tm}((\Gamma, \Gamma_*), \Delta_*[\sigma \circ \mathrm{w}]) \tag{2.6.41}$$
$$\_{}_* \;:\; (t : \mathrm{Tm}(\Gamma, A)) \longrightarrow \mathrm{Tm}((\Gamma, \Gamma_*), A_*[\mathrm{id}, t[\mathrm{w}]]) \tag{2.6.42}$$

Such that for the category structure we have:

$$\mathrm{id}_* \;=\; \mathrm{v} \tag{2.6.43}$$
$$(\sigma \circ \delta)_* \;=\; \sigma_*[\delta \circ \mathrm{w}, \delta_*] \tag{2.6.44}$$

for substitutions with $A : \mathrm{Ty}(\Gamma)$ and $t : \mathrm{Tm}(\Gamma, A)$ we have:

$$(A[\sigma])_* \;=\; A_*[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}] \tag{2.6.45}$$
$$(t[\sigma])_* \;=\; t_*[\sigma \circ \mathrm{w}, \sigma_*] \tag{2.6.46}$$

for both the terminal object and unit types:

$$\top_* \;=\; \top \tag{2.6.47}$$
$$\epsilon_* \;=\; \epsilon \tag{2.6.48}$$

for context comprehension:

$$(\Gamma, A)_* \;=\; \Sigma(\Gamma_*[\mathrm{w}], A_*[\mathrm{w}^2, \mathrm{v}, \mathrm{v}[\mathrm{w}]]) \tag{2.6.49}$$
$$(\sigma, t)_* \;=\; (\sigma_*, t_*) \tag{2.6.50}$$
$$\pi_1(\sigma)_* \;=\; \pi_1(\sigma_*) \tag{2.6.51}$$
$$\pi_2(\sigma)_* \;=\; \pi_2(\sigma_*) \tag{2.6.52}$$

for product types:

$$\Sigma(A, B)_* \;=\; \Sigma(A_*[\eta_1], B_*[\eta_2]) \tag{2.6.53}$$
$$(s, t)_* \;=\; (s_*, t_*) \tag{2.6.54}$$
$$\pi_1(t)_* \;=\; \pi_1(t_*) \tag{2.6.55}$$
$$\pi_2(t)_* \;=\; \pi_2(t_*) \tag{2.6.56}$$

where:

$$\eta_1 \;=\; (\mathrm{w}, \pi_1(\mathrm{v})) \tag{2.6.57}$$
$$\eta_2 \;=\; (\mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]) \tag{2.6.58}$$

Remark 2.6.8. A parametric category with families needs product and unit types, so that we can define $\top_*$ for $\top$ the empty context, and $(\Gamma, A)_*$.

Remark 2.6.9. To treat binary parametricity, a product for contexts should be added, so that for $\Gamma : \mathrm{Ob}$ we can define:

$$\Gamma_* \;:\; \mathrm{Ty}(\Gamma, \Gamma) \tag{2.6.59}$$

We could assume democracy, as together with product types it implies such a product of contexts.

The main point of this lengthy definition is that:

Proposition 2.6.10. *The theory of parametric category with families is an extension by section of the theory of category with families with unit and product types.*

PROOF. It is clear that $\_{*}$ is inductively defined on all operations. We need to check that this respects the equations of categories with families. A direct proof can be found in Section A.1. □

Example 2.6.11. For categories with families, Proposition 2.6.10 and Theorem 2.3.3 imply that the forgetful functor from parametric categories with families to categories with families has a right adjoint R (ignoring the unary/binary discrepancy).

We denote by $\mathbb{I}_X$ the parametric category with families freely generated by an object X. Then for $\mathcal{C}$ a category with families, we have:

$$\begin{aligned} \mathrm{Ob}_{R(\mathcal{C})} &= \mathrm{Hom}_{\mathrm{Set}}(\{X\}, \mathrm{Ob}_{R(\mathcal{C})}) && (2.6.60)\\ &= \mathrm{Hom}_{\mathrm{pCwF}}(\mathbb{I}_X, R(\mathcal{C})) && (2.6.61)\\ &= \mathrm{Hom}_{\mathrm{CwF}}(U(\mathbb{I}_X), \mathcal{C}) && (2.6.62)\end{aligned}$$

The category with families $U(\mathbb{I}_X)$ should be isomorphic to the (non-parametric) category with families freely generated by:

$$\begin{aligned} X &: \mathrm{Ob} && (2.6.63)\\ X_* &: \mathrm{Ty}(x_0, x_1 : X) && (2.6.64)\\ X_{**} &: \mathrm{Ty}(x_{00}, x_{10}, x_{01}, x_{11} : X,\\ &\qquad X_*(x_{00}, x_{01}), X_*(x_{00}, x_{01}), X_*(x_{10}, x_{11}), X_*(x_{01}, x_{11})) && (2.6.65)\\ &\vdots \end{aligned}$$

So giving a context in $R(\mathcal{C})$ is equivalent to giving:

$$\begin{aligned} \Gamma &: \mathrm{Ob}_{\mathcal{C}} && (2.6.66)\\ \Gamma_* &: \mathrm{Ty}_{\mathcal{C}}(x_0, x_1 : \Gamma) && (2.6.67)\\ \Gamma_{**} &: \mathrm{Ty}_{\mathcal{C}}(x_{00}, x_{10}, x_{01}, x_{11} : \Gamma,\\ &\qquad \Gamma_*(x_{00}, x_{01}), \Gamma_*(x_{00}, x_{01}), \Gamma_*(x_{10}, x_{11}), \Gamma_*(x_{01}, x_{11})) && (2.6.68)\\ &\vdots \end{aligned}$$

with similar formulas for types, terms, and so on. This is intuitively a semi-cubical type, although we do not make this formal.

2.7. Extending external parametricity to arrow types and a universe

We want to extend the standard parametricity from Section 2.6 to arrow types and a universe. First we define them using a signature.

Definition 2.7.1. Arrow types for a category with families consist of:

$$\begin{aligned} \Pi &: (A : \mathrm{Ty}(\Gamma)) \to \mathrm{Ty}(\Gamma, A) \to \mathrm{Ty}(\Gamma) && (2.7.1)\\ \mathrm{ap} &: \mathrm{Tm}(\Gamma, \Pi(A, B)) \to \mathrm{Tm}((\Gamma, A), B) && (2.7.2)\\ \lambda &: \mathrm{Tm}((\Gamma, A), B) \to \mathrm{Tm}(\Gamma, \Pi(A, B)) && (2.7.3)\end{aligned}$$

such that we have:

$$\mathrm{ap}(\lambda(t)) = t \tag{2.7.4}$$
$$\lambda(\mathrm{ap}(t)) = t \tag{2.7.5}$$

with equations for substitutions:

$$\Pi(A, B)[\sigma] = \Pi(A[\sigma], B[\sigma \circ \mathrm{w}, \mathrm{v}]) \tag{2.7.6}$$
$$\lambda(t)[\sigma] = \lambda(t[\sigma \circ \mathrm{w}, \mathrm{v}]) \tag{2.7.7}$$

Remark 2.7.2. The previous definition implies that:

$$\mathrm{ap}(t)[\sigma \circ \mathrm{w}, \mathrm{v}] = \mathrm{ap}(t[\sigma]) \tag{2.7.8}$$

Definition 2.7.3. A universe for a category with families consists of:

$$\mathcal{U} : \mathrm{Ty}(\Gamma) \tag{2.7.9}$$
$$\mathrm{El} : \mathrm{Tm}(\Gamma, \mathcal{U}) \to \mathrm{Ty}(\Gamma) \tag{2.7.10}$$
$$\top_{\mathcal{U}} : \mathrm{Tm}(\Gamma, \mathcal{U}) \tag{2.7.11}$$
$$\Sigma_{\mathcal{U}} : (A : \mathrm{Tm}(\Gamma, \mathcal{U})) \to \mathrm{Tm}((\Gamma, \mathrm{El}(A)), \mathcal{U}) \to \mathrm{Tm}(\Gamma, \mathcal{U}) \tag{2.7.12}$$
$$\Pi_{\mathcal{U}} : (A : \mathrm{Tm}(\Gamma, \mathcal{U})) \to \mathrm{Tm}((\Gamma, \mathrm{El}(A)), \mathcal{U}) \to \mathrm{Tm}(\Gamma, \mathcal{U}) \tag{2.7.13}$$

such that we have:

$$\mathrm{El}(\top_{\mathcal{U}}) = \top \tag{2.7.14}$$
$$\mathrm{El}(\Sigma_{\mathcal{U}}(A, B)) = \Sigma(\mathrm{El}(A), \mathrm{El}(B)) \tag{2.7.15}$$
$$\mathrm{El}(\Pi_{\mathcal{U}}(A, B)) = \Pi(\mathrm{El}(A), \mathrm{El}(B)) \tag{2.7.16}$$

with equations for substitutions:

$$\mathcal{U}[\sigma] = \mathcal{U} \tag{2.7.17}$$
$$\mathrm{El}(A)[\sigma] = \mathrm{El}(A[\sigma]) \tag{2.7.18}$$
$$\top_{\mathcal{U}}[\sigma] = \top_{\mathcal{U}} \tag{2.7.19}$$
$$\Sigma_{\mathcal{U}}(A, B)[\sigma] = \Sigma_{\mathcal{U}}(A[\sigma], B[\sigma \circ \mathrm{w}, \mathrm{v}]) \tag{2.7.20}$$
$$\Pi_{\mathcal{U}}(A, B)[\sigma] = \Pi_{\mathcal{U}}(A[\sigma], B[\sigma \circ \mathrm{w}, \mathrm{v}]) \tag{2.7.21}$$

Now we are ready to extend parametricity to arrow types and a universe.

Definition 2.7.4. A category with families with arrow types and a universe is called parametric if it is parametric as a category with families (see Definition 2.6.7), such that for arrow types we have:

$$\Pi(A, B)_* = \Pi(A[\sigma_1], \Pi(A_*[\sigma_2], B_*[\sigma_3])) \tag{2.7.22}$$
$$\mathrm{ap}(t)_* = (\mathrm{ap}(\mathrm{ap}(t_*))[\nu_1] \tag{2.7.23}$$
$$\lambda(t)_* = \lambda(\lambda(t_*[\nu_2])) \tag{2.7.24}$$

where:

$$\sigma_1 = \mathrm{w}^2 \tag{2.7.25}$$
$$\sigma_2 = (\mathrm{w}^2, \mathrm{v}) \tag{2.7.26}$$
$$\sigma_3 = (\mathrm{w}^4, \mathrm{v}[\mathrm{w}], (\mathrm{v}[\mathrm{w}^3], \mathrm{v}), (\mathrm{ap}(\mathrm{v}))[\mathrm{w}]) \tag{2.7.27}$$
$$\nu_1 = (\mathrm{w}^2, \pi_1(\mathrm{v}), \mathrm{v}[\mathrm{w}], \pi_2(\mathrm{v})) \tag{2.7.28}$$
$$\nu_2 = (\mathrm{w}^3, \mathrm{v}[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v})) \tag{2.7.29}$$

and for the universe we have:

$$\begin{aligned}
\mathcal{U}_* &= \Pi(\mathrm{El}(\mathrm{v}), \mathcal{U}) && (2.7.30)\\
\mathrm{El}(A)_* &= \mathrm{El}(\mathrm{ap}(A_*)) && (2.7.31)\\
(\top_{\mathcal{U}})_* &= \lambda(\top_{\mathcal{U}}) && (2.7.32)\\
\Sigma_{\mathcal{U}}(A, B)_* &= \lambda(\Sigma_{\mathcal{U}}(\mathrm{ap}(A_*)[\eta_1], \mathrm{ap}(B_*)[\eta_2])) && (2.7.33)\\
\Pi_{\mathcal{U}}(A, B)_* &= \lambda(\Pi_{\mathcal{U}}(A[\sigma_1], \Pi_{\mathcal{U}}(\mathrm{ap}(A_*)[\sigma_2], \mathrm{ap}(B_*)[\sigma_3]) && (2.7.34)
\end{aligned}$$

where:

$$\begin{aligned}
\eta_1 &= (\mathrm{w}, \pi_1(\mathrm{v})) && (2.7.35)\\
\eta_2 &= (\mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]) && (2.7.36)
\end{aligned}$$

Now an important result from [**Moe21**], with the same proof:

Proposition 2.7.5. *Parametricity is an extension by section of categories with families with arrow types and a universe.*

PROOF. We need to check that the given inductive definitions respect the equations for arrow types and universes. This is checked in Section A.2. ☐

Remark 2.7.6. Being a parametric category with families with arrow types and a universe is restrictive. For example it is well-known that it contradicts the law of excluded middle, using a two-point type $\mathbb{B}$ with $0, 1 : \mathbb{B}$.

Indeed using that:

$$X = \mathbb{B} \quad \vee \quad X \neq \mathbb{B} \tag{2.7.37}$$

we can define:

$$\psi \quad : \quad \Pi(X : \mathcal{U}).\, X \to X \tag{2.7.38}$$

by:

$$\begin{aligned}
\psi(\mathbb{B}, 0) &= 1 && (2.7.39)\\
\psi(\mathbb{B}, 1) &= 0 && (2.7.40)
\end{aligned}$$

and:

$$\psi(X, x) \quad = \quad x \tag{2.7.41}$$

when $X \neq \mathbb{B}$. But parametricity implies that given a term:

$$f \quad : \quad \Pi(X : \mathcal{U}).\, X \to X \tag{2.7.42}$$

we have another term:

$$f_* \quad : \quad \Pi(X : \mathcal{U},\ X_* : X \to \mathcal{U},\ x : X).\, X_*(x) \to X_*(f(x)) \tag{2.7.43}$$

So that for any:

$$P \quad : \quad \mathbb{B} \to \mathcal{U} \tag{2.7.44}$$

we have:

$$\psi_*(\mathbb{B}, P, 0) \quad : \quad P(0) \to P(1) \tag{2.7.45}$$

which is incoherent.

Remark 2.7.7. An unpleasant consequence of Remark 2.7.6 is that freely adding parametricity to sets leads to an incoherent model (assuming the law of excluded middle for sets). So the left adjoint to the functor forgetting parametricity can send a coherent model to an incoherent one.

The right adjoint does not suffer from the same defect, at least when we have an empty type $\bot$. Indeed, using the counit:

$$\epsilon_{\mathcal{C}} \quad : \quad UR(\mathcal{C}) \to \mathcal{C} \tag{2.7.46}$$

we know that:

$$t \quad : \quad \mathrm{Tm}_{UR(\mathcal{C})}(\top, \bot) \tag{2.7.47}$$

gives:

$$\epsilon_{\mathcal{C}}(t) \quad : \quad \mathrm{Tm}_{\mathcal{C}}(\top, \bot) \tag{2.7.48}$$

so that $UR(\mathcal{C})$ incoherent implies $\mathcal{C}$ incoherent.

We can define *internal* parametricity for category with families with product and unit types by adding:

$$\begin{aligned}
\mathrm{refl} \quad &: \quad (\Gamma : \mathrm{Ob}) \to \mathrm{Tm}(\Gamma, \Gamma_*) &(2.7.49)\\
\mathrm{refl} \quad &: \quad (A : \mathrm{Ty}(\Gamma)) \to \mathrm{Tm}((\Gamma, A), A_*[\mathrm{w}, \mathrm{refl}_\Gamma[\mathrm{w}], \mathrm{v}]) &(2.7.50)\\
_ \quad &: \quad (t : \mathrm{Tm}(\Gamma, A)) \to t_*[\mathrm{id}, \mathrm{refl}_\Gamma] = \mathrm{refl}_A[\mathrm{id}, t] &(2.7.51)\\
_ \quad &: \quad (\sigma : \mathrm{Hom}(\Gamma, \Delta)) \to \sigma_*[\mathrm{id}, \mathrm{refl}_\Gamma] = \mathrm{refl}_\Delta[\sigma] &(2.7.52)
\end{aligned}$$

with equations defining them inductively on any constructors. But when trying to extend this to arrow types and universes, we can't find any valid inductive definition for:

$$\begin{aligned}
\mathrm{refl}_{\Pi(A,B)} \quad &= \quad ? &(2.7.53)\\
\mathrm{refl}_{\mathrm{El}(A)} \quad &= \quad ? &(2.7.54)
\end{aligned}$$

We conjecture that it is not possible to find such a definition.

Remark 2.7.8. We give a sketch of proof that in our framework, internal parametricity for categories cannot be extended to exponentials.

Let $\mathcal{R}$ be the category classifying reflexive graphs, i.e. the category such that $\mathcal{C}^{\mathcal{R}}$ is the category of reflexive graphs in $\mathcal{C}$. Then the forgetful functor from internally parametric categories to categories is the categorical extension by section:

$$\mathrm{CoAlg}_{\mathrm{Cat}}(E, d) \to \mathrm{Cat} \tag{2.7.55}$$

induced by:

$$\begin{aligned}
E \quad &: \quad \mathrm{Cat} \to \mathrm{Cat} &(2.7.56)\\
E(\mathcal{C}) \quad &= \quad \mathcal{C}^{\mathcal{R}} &(2.7.57)
\end{aligned}$$

with $d_{\mathcal{C}}$ sending a reflexive graph to its object of vertices.

Assume that it is possible to extend internal parametricity for categories to exponentials. Then for $\mathcal{C}$ a category with exponentials, we expect that $\mathcal{C}^{\mathcal{R}}$ would have exponentials as well, and $d_{\mathcal{C}}$ would preserve them. A precise syntactic argument toward this result seems delicate to write down, but intuitively plausible. Given:

$$X, Y \quad : \quad \mathcal{C}^{\mathcal{R}} \tag{2.7.58}$$

a vertex in:

$$Y^X \quad : \quad \mathcal{C}^{\mathcal{R}} \tag{2.7.59}$$

should not be any map between vertices, but a map preserving edges, as explained for example in Section 4.4 of [AGJ14]. This precisely contradicts $d_{\mathcal{C}}$ commuting with exponentials.

A similar argument should be doable for a universe, although more involved as multiple choices of universe are possible, whereas exponentials are fixed up to isomorphisms by their definition.

In Chapter 3 we will only consider models without arrow types nor a universe.

2.8. Conjectural examples

In this section we discuss two potential extensions by section related to univalence. The first is *setoid type theory* as in [ABKT19] seen as a 1-truncated form of univalence. The second is *univalent parametricity* from [TTS21], where a univalent universe is assumed to begin with.

- Setoid type theory [ABKT19] is a type theory with two sorts of types called sets and propositions, denoted by:

$$\Gamma \quad \vdash_S \quad A \tag{2.8.1}$$

$$\Gamma \quad \vdash_P \quad A \tag{2.8.2}$$

  We give some of the rules of this type theory using a type-theoretic notation. For example any context comes with a relation:

$$\frac{\Gamma \ \vdash}{\Gamma,\, \Gamma \ \vdash_P \ \Gamma_*} \tag{2.8.3}$$

  and this relation is reflexive:

$$\frac{\Gamma \ \vdash}{x : \Gamma \ \vdash_P \ \mathrm{refl}_\Gamma : \Gamma_*[x, x]} \tag{2.8.4}$$

  Any set comes with a heterogeneous relation:

$$\frac{\Gamma \ \vdash_S \ A}{x_0, x_1 : \Gamma,\ \Gamma_*[x_0, x_1],\ A[x_0],\ A[x_1] \ \vdash_P \ A_*} \tag{2.8.5}$$

  and this relation is reflexive:

$$\frac{\Gamma \ \vdash_S \ A}{x : \Gamma,\ y : A \ \vdash_P \ \mathrm{refl}_A : A_*[x, x, \mathrm{refl}_\Gamma, y, y]} \tag{2.8.6}$$

  Any proposition or set comes with coercions, similar to transports, so that for $\epsilon = S$ or P we have:

$$\frac{\Gamma \ \vdash_\epsilon \ A}{x_0, x_1 : \Gamma,\ \Gamma_*[x_0, x_1],\ A[x_0] \ \vdash_\epsilon \ \overrightarrow{\mathrm{coe}}_A : A[x_1]} \tag{2.8.7}$$

  and:

$$\frac{\Gamma \ \vdash_\epsilon \ A}{x_0, x_1 : \Gamma,\ \Gamma_*[x_0, x_1],\ A[x_1] \ \vdash_\epsilon \ \overleftarrow{\mathrm{coe}}_A : A[x_0]} \tag{2.8.8}$$

  We also assume product and unit types for sets and propositions, as well as arrow types for propositions and a set of propositions.

Remark 2.8.1. Arrow types for sets and a universe of sets from [**ABKT19**] do not fit into this framework, due to their issues with reflexivities. See Remark 2.7.8 for more details.

We conjecture that this restricted setoid type theory (without arrow types for sets, or a set of sets) is an extension by section of type theory with two sorts of types. We give a few examples of inductive definitions:

- For arrow types for proposition we define:

$$\overrightarrow{\mathrm{coe}}_{A\to B}[f] \quad = \quad \lambda x.\, \overrightarrow{\mathrm{coe}}_B[f(\overleftarrow{\mathrm{coe}}_A[x])] : A_1 \to B_1 \tag{2.8.9}$$

- For the set of proposition:

$$\vdash_S \quad \mathrm{Prop} \tag{2.8.10}$$

$$x : \mathrm{Prop} \quad \vdash_P \quad \mathrm{El}[x] \tag{2.8.11}$$

we define:

$$\mathrm{Prop}_*[x_0, x_1] \quad = \quad (\mathrm{El}[x_0] \to \mathrm{El}[x_1]) \times (\mathrm{El}[x_1] \to \mathrm{El}[x_0]) \tag{2.8.12}$$

and:

$$\overrightarrow{\mathrm{coe}}_{\mathrm{Prop}}[x_0, x_1, x_*, y_0] \quad = \quad \pi_1(x_*)(y_0) : \mathrm{El}[x_1] \tag{2.8.13}$$

We omit the many other necessary equations.

Remark 2.8.2. Reflexivities and coercions imply that Γ_* is reflexive, symmetric and transitive, justifying the name of setoid type theory.

Remark 2.8.3. Despite the problem with arrow types and reflexivities, arrow types for propositions work fine because only sets have reflexivities, not propositions. A set of propositions works fine because we can define:

$$\mathrm{refl}_x \quad = \quad (\mathrm{id}_x, \mathrm{id}_x) \tag{2.8.14}$$

in:

$$\mathrm{Prop}_*[x, x] \quad = \quad (\mathrm{El}[x] \to \mathrm{El}[x]) \times (\mathrm{El}[x] \to \mathrm{El}[x]) \tag{2.8.15}$$

where:

$$\mathrm{id}_x \quad = \quad \lambda(y : \mathrm{El}[x]).\, y \tag{2.8.16}$$

Starting from the model where sets are actual sets and propositions are sub-singletons, applying the right adjoint R should build the usual setoid model.

- In [**TTS21**] a variant of parametricity called *univalent parametricity* is introduced. It starts from type theory with identity types and a univalent universe. In this theory any type comes with an inductively defined relation *equivalent to its identity type*. This allows to circumvent the troubles with arrow types and universes, indeed:
  - For arrow types, the relation is defined by:

$$(A \to B)_*[f_0, f_1] \quad = \quad \Pi(x_0, x_1 : A).\, A_*[x_0, x_1] \to B_*[f_0(x_0), f_1(x_1)] \tag{2.8.17}$$

  Then we need to prove that if:

$$A_*[x_0, x_1] \quad \simeq \quad x_0 =_A x_1 \tag{2.8.18}$$

$$B_*[y_0, y_1] \quad \simeq \quad y_0 =_B y_1 \tag{2.8.19}$$

then:

$$(A \longrightarrow B)_*[f_0, f_1] \quad \simeq \quad f_0 =_{A \to B} f_1 \tag{2.8.20}$$

This uses function extensionality, which holds because we have a univalent universe.

- For the universe we define:

$$\mathcal{U}_*[A_0, A_1] \quad = \quad A_0 \simeq A_1 \tag{2.8.21}$$

The requirement that:

$$\mathcal{U}_*[A_0, A_1] \quad \simeq \quad A_0 =_{\mathcal{U}} A_1 \tag{2.8.22}$$

is precisely univalence.

We conjecture that this incomplete description is part of an interpretation. This seems to avoid assuming reflexivities by building them from reflexivities in the identity types.

This conjecture implies a right adjoint R. For $\mathcal{C}$ a model of univalent type theory, a type in $R(\mathcal{C})$ should be a type X in $\mathcal{C}$ with a semi-cubical structure on X, together with a proof that this semi-cubical structure is equivalent to the one induced by the iterated identity types.

Moreover we conjecture that it is possible to use $R(\mathcal{C})$ as an intermediary model toward building a model equivalent to $\mathcal{C}$ enjoying *univalence by definition*, meaning that we have:

$$(A =_{\mathcal{U}} B) \quad \equiv \quad (A \simeq B) \tag{2.8.23}$$

where $\_ \equiv \_$ is definitional equality.

CHAPTER 3

Notions of parametricity as monoidal models

It is not straightforward to prove that cofreely parametric models are indeed cubical with the approach presented in Chapter 2, because its description for the right adjoints are not convenient to work with. In this chapter we give an alternative to extensions by section, which gives compact descriptions for freely and cofreely parametric objects.

This alternative is remarkably simple. We assume a symmetric monoidal closed category $\mathcal{V}$ of models of type theory to begin with. We define a notion of parametricity $\mathcal{M}$ simply as a monoid in $\mathcal{V}$, that is a monoidal model of type theory. Then we define $\mathcal{M}$-parametric models as $\mathcal{M}$-modules. Finally we get very compact descriptions for the left and right adjoints to functors forgetting parametricity using induced and coinduced modules.

We need symmetric monoidal closed categories of models of type theory. We will use the following examples:

- The category of categories is cartesian closed. This will be used as a guiding example throughout this chapter.
- The category of lex categories is symmetric monoidal closed, using the naive exponential for lex categories. For this to work, we have to use a strict variant of lex category.
- The category of clans is symmetric monoidal closed. As for lex categories, we use a strict variant of clan.

While we have to consider strict variants in order to stay in a 1-categorical setting, a better solution would be to work with the 2-categories of lex categories and clans. This is left for further work.

Remark 3.0.1. We conjecture that any notion of model of type theory with product and unit types (e.g. category with families, natural models...) gives a symmetric monoidal closed category of models, at least when appropriately strictified (or better yet, they form a symmetric monoidal closed 2-category). On the other hand we do not expect models of type theory with arrow types or a universe to form such a category, because of their issues with internal notions of parametricity.

This chapter is organized as follows:

- In Section 3.1 we go back to the definition of parametric categories from Section 1.1 and reformulate it using monoidal categories. This motivates the abstract definitions in the next section.
- In Section 3.2 we assume a symmetric monoidal closed category $\mathcal{V}$. We define notions of parametricity for $\mathcal{V}$ as monoids in $\mathcal{V}$, and parametric objects as modules. Then we give many examples of notion of parametricity for categories.

- In Section 3.3 we prove that functors forgetting parametricity in the sense of Section 3.2 have left and right adjoints, and give explicit descriptions for these. We explain how to obtain any category of cubical objects using these right adjoints.
- In Section 3.4 we prove that the category of lex categories is symmetric monoidal closed (using a strict variant of lex categories). Then in Section 3.5 we use Section 3.2 to define notions of parametricity for lex categories. In Section 3.6 we show that lex categories of truncated semi-cubical (or cubical with reflexivities) objects are cofreely parametric.
- In Section 3.7 we define the strict variant of clans. Then in Section 3.8 we show that the category of strict clans is symmetric monoidal closed. Finally, in Section 3.9 we use Section 3.2 to define notions of parametricity for strict clans and we show that clans of Reedy fibrant semi-cubical (or cubical with reflexivities) objects are cofreely parametric.

3.1. Parametricity for categories revisited

This preliminary section explains the reasoning leading to the abstract axiomatisation of the next section. We consider plain categories as models for type theory, and examine parametric categories very closely. Recall the standard parametricity for categories:

Definition 3.1.1. A parametric category is a category $\mathcal{C}$ equipped with:

- An endofunctor:
$$\_{}_* \quad : \quad \mathcal{C} \to \mathcal{C} \tag{3.1.1}$$
- For any $\Gamma : \mathcal{C}$ two morphisms:
$$d^0_\Gamma, d^1_\Gamma \quad : \quad \Gamma_* \to \Gamma \tag{3.1.2}$$
natural in Γ.

Recall that given an object Γ in a parametric category $\mathcal{C}$, we can iterate $\_{}_*$, building the following diagram:

$$\begin{tikzcd}
\Gamma & \Gamma_* \arrow[l, shift left, "d^0_\Gamma"'] \arrow[l, shift right, "d^1_\Gamma"] & \Gamma_{**} \arrow[l, shift left, "d^0_{\Gamma_*}"'] \arrow[l, shift right, "d^1_{\Gamma_*}"] \arrow[l, bend right=60, "(d^0_\Gamma)_*"'] \arrow[l, bend left=60, "(d^1_\Gamma)_*"] & \cdots
\end{tikzcd}$$

We will now prove that this is a semi-cubical object in $\mathcal{C}$ (meaning a cubical object without reflexivities) with Γ as its object of points. We give an auxiliary definition, which will be useful to describe semi-cubical objects:

Definition 3.1.2. Let $\square$ be the strict monoidal category freely generated by:

- An object $\mathbb{I}$.
- Two morphisms:
$$d^0, d^1 \quad : \quad \mathbb{I} \to 1 \tag{3.1.3}$$
where 1 is the monoidal unit.

The reader unfamiliar with semi-cubical objects can take the following as a definition:

Proposition 3.1.3. *A functor from* $\square$ *to* $\mathcal{C}$ *is a semi-cubical object in* $\mathcal{C}$.

Remark 3.1.4. It is standard to define semi-cubical objects as presheaves on $\square^{op}$. We reverse arrows here to avoid including opposites in our axiomatisation. We will give remarks whenever this could lead to confusion.

Remark 3.1.5. Objects in $\square$ are of the form:

$$\mathbb{I} \otimes \cdots \otimes \mathbb{I} \tag{3.1.4}$$

and morphisms are tensors of d^0, d^1 and $\mathrm{id}_{\mathbb{I}}$.

For example morphisms in:

$$\mathbb{I} \otimes \mathbb{I} \otimes \mathbb{I} \quad \to \quad \mathbb{I} \otimes \mathbb{I} \tag{3.1.5}$$

are precisely:

$$d^{\epsilon} \otimes \mathrm{id}_{\mathbb{I}} \otimes \mathrm{id}_{\mathbb{I}} \tag{3.1.6}$$
$$\mathrm{id}_{\mathbb{I}} \otimes d^{\epsilon} \otimes \mathrm{id}_{\mathbb{I}} \tag{3.1.7}$$
$$\mathrm{id}_{\mathbb{I}} \otimes \mathrm{id}_{\mathbb{I}} \otimes d^{\epsilon} \tag{3.1.8}$$

where $\epsilon = 0, 1$.

There is a clear analogy between the natural transformations:

$$d^0, d^1 \quad : \quad \__* \to \mathrm{Id} \tag{3.1.9}$$

in a parametric category and the generator for semi-cubes:

$$d^0, d^1 \quad : \quad \mathbb{I} \to 1 \tag{3.1.10}$$

To make this precise we need auxiliary definitions. The first one is very well-known:

Definition 3.1.6. For $\mathcal{C}$ a category, we define $\mathcal{E}nd_{\mathcal{C}}$ as the strict monoidal category of endofunctors of $\mathcal{C}$, with composition as tensor and identity as unit.

The second definition is not as common, but very elementary.

Definition 3.1.7. Let $\mathcal{M}$ be a strict monoidal category. An $\mathcal{M}$-module consists of:

- A category $\mathcal{C}$.
- A strict monoidal functor:

$$\mathcal{M} \to \mathcal{E}nd_{\mathcal{C}} \tag{3.1.11}$$

This is just the usual definition of a monoid action, valid in any cartesian closed category, specialised to the category of categories.

Remark 3.1.8. We could equivalently define an $\mathcal{M}$-module as a category $\mathcal{C}$ with a functor:

$$\_ \otimes \_ \quad : \quad \mathcal{M} \times \mathcal{C} \to \mathcal{C} \tag{3.1.12}$$

such that for all $i, j : \mathcal{M}$ and $\Gamma : \mathcal{C}$ we have:

$$(i \otimes j) \otimes \Gamma \quad = \quad i \otimes (j \otimes \Gamma) \tag{3.1.13}$$
$$1 \otimes \Gamma \quad = \quad \Gamma \tag{3.1.14}$$

functorially in i, j and Γ.

Now we can state:

Lemma 3.1.9. *Giving a parametric category is equivalent to giving a $\square$-module.*

Proof. By Proposition 3.1.2 giving a monoidal functor:

$$\square \to \mathcal{D} \tag{3.1.15}$$

is equivalent to giving:

$$\begin{aligned} \Gamma &: \mathcal{D} && (3.1.16)\\ d^0, d^1 &: \Gamma \to 1 && (3.1.17)\end{aligned}$$

So a monoidal functor:

$$\square \to \mathcal{E}nd_{\mathcal{C}} \tag{3.1.18}$$

is equivalent to:

$$\begin{aligned} \_{*} &: \mathcal{E}nd_{\mathcal{C}} && (3.1.19)\\ d^0, d^1 &: \mathrm{Hom}_{\mathcal{E}nd_{\mathcal{C}}}(\_{*}, \mathrm{Id}) && (3.1.20)\end{aligned}$$

□

Remark 3.1.10. For any $\square$-module $\mathcal{C}$ with an object Γ, we have a functor:

$$\begin{aligned} F &: \square \to \mathcal{C} && (3.1.21)\\ F(i) &= i \otimes \Gamma && (3.1.22)\end{aligned}$$

such that:

$$F(1) = \Gamma \tag{3.1.23}$$

So Γ is indeed the object of points of a semi-cubical object in $\mathcal{C}$.

3.2. Notions of parametricity as monoids

Now, using insights from the first section, we will define an abstract framework for parametricity. We will use categories as an example throughout to give intuitions on definitions and results presented here.

Notation 3.2.1. We assume $\mathcal{V}$ a symmetric monoidal closed category, with:

- A tensor product:

$$\_ \otimes \_ : \mathcal{V} \to \mathcal{V} \to \mathcal{V} \tag{3.2.1}$$

- A unit $1 : \mathcal{V}$.
- An arrow:

$$\_ \multimap \_ : \mathcal{V} \to \mathcal{V} \to \mathcal{V} \tag{3.2.2}$$

An object in $\mathcal{V}$ should be thought of as a model of type theory.

Example 3.2.2. We can take the category of categories as $\mathcal{V}$. Its monoidal closed structure is in fact cartesian.

Example 3.2.3. We can take the category of abelian groups as $\mathcal{V}$. Its tensor is not a cartesian product.

Now we give our main definition:

Definition 3.2.4. A notion of parametricity for $\mathcal{V}$ is a monoid in $\mathcal{V}$.

Next remark makes this definition fully explicit.

Remark 3.2.5. A monoid in $\mathcal{V}$ consists of $\mathcal{M} : \mathcal{V}$ with:

$$\mu \quad : \quad \mathcal{M} \otimes \mathcal{M} \to \mathcal{M} \tag{3.2.3}$$

$$\epsilon \quad : \quad 1 \to \mathcal{M} \tag{3.2.4}$$

such that the following diagrams commute:

$$\begin{tikzcd}
(\mathcal{M} \otimes \mathcal{M}) \otimes \mathcal{M} \arrow[r, "\mu\otimes\mathcal{M}"] \arrow[dd, "\cong"'] & \mathcal{M} \otimes \mathcal{M} \arrow[dr, "\mu"] & \\
 & & \mathcal{M} \\
\mathcal{M} \otimes (\mathcal{M} \otimes \mathcal{M}) \arrow[r, "\mathcal{M}\otimes\mu"'] & \mathcal{M} \otimes \mathcal{M} \arrow[ur, "\mu"'] &
\end{tikzcd}$$

$$\begin{tikzcd}
\mathcal{M} \arrow[r, "\mathrm{id}"] \arrow[d, "\cong"'] & \mathcal{M} \\
\mathcal{M} \otimes 1 \arrow[r, "\mathcal{M}\otimes\epsilon"'] & \mathcal{M} \otimes \mathcal{M} \arrow[u, "\mu"']
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathcal{M} \arrow[r, "\mathrm{id}"] \arrow[d, "\cong"'] & \mathcal{M} \\
1 \otimes \mathcal{M} \arrow[r, "\epsilon\otimes\mathcal{M}"'] & \mathcal{M} \otimes \mathcal{M} \arrow[u, "\mu"']
\end{tikzcd}$$

Remark 3.2.6. Here diagrams are required to commute up to equality. So if $\mathcal{V}$ is the cartesian closed category of categories, a monoid in $\mathcal{V}$ is a *strict* monoidal category.

Example 3.2.7. There are many examples of notions of parametricity for categories besides $\square$ from Proposition 3.1.2. Indeed all category of cubes from [**BM17**] are monoidal. So we have notions of parametricity for categories corresponding to all kinds of cubes:

- We can have symmetries, diagonals or reflexivities. For readers familiar with cubical type theories, they correspond to structural rules on interval variables.
- We can have connections or inverses.

Many more variants are possible, for example:

- We can consider the monoidal category freely generated by an object $\mathbb{I}$ and:

$$d^0, \cdots, d^n \quad : \quad \mathbb{I} \to 1 \tag{3.2.5}$$

  This gives an n-ary notions of parametricity, where any type comes with an n-ary predicates rather than a relation. For $n = 1$ we get unary parametricity, for $n = 2$ we get the standard parametricity.
- We can also consider a monoidal category generated by several objects. For example semi-bicubes form a monoidal category freely generated by two objects $\mathbb{I}$ and $\mathbb{J}$ with:

$$d^0, d^1 \quad : \quad \mathbb{I} \to 1 \tag{3.2.6}$$

$$e^0, e^1 \quad : \quad \mathbb{J} \to 1 \tag{3.2.7}$$

Remark 3.2.8. The fact that we used the opposite from the standard category of semi-cubes is not an issue here, as the opposite of a monoidal category is monoidal.

Remark 3.2.9. The monoidal category corresponding to unary parametricity is freely generated by an object $\mathbb{I}$ and a map:

$$d \quad : \quad \mathbb{I} \to 1 \tag{3.2.8}$$

It is the (opposite of the) category of augmented semi-simplices.

There is a similar result for the category of augmented simplices, which is freely generated by a comonoid $\mathbb{I}$.

Now we can adapt our definition of parametric categories.

Remark 3.2.10. For $\mathcal{C} : \mathcal{V}$ we have a monoid $\mathcal{E}nd_{\mathcal{C}}$ in $\mathcal{V}$ such that:

- Its underlying object is:

$$\mathcal{C} \multimap \mathcal{C} \tag{3.2.9}$$

- Its product is composition.
- Its unit is the identity.

Definition 3.2.11. An object $\mathcal{C} : \mathcal{V}$ is called $\mathcal{M}$-parametric if it is an $\mathcal{M}$-module, i.e. if we are given a morphism of monoid:

$$\mathcal{M} \to \mathcal{E}nd_{\mathcal{C}} \tag{3.2.10}$$

Next remark gives an explicit reformulation for this.

Remark 3.2.12. An $\mathcal{M}$-module structure on an object $\mathcal{C}$ is equivalent to:

$$\alpha \quad : \quad \mathcal{M} \otimes \mathcal{C} \to \mathcal{C} \tag{3.2.11}$$

such that the following diagrams commute:

$$\begin{tikzcd}
(\mathcal{M} \otimes \mathcal{M}) \otimes \mathcal{C} \arrow[r, "\mu\otimes\mathcal{C}"] \arrow[d, "\cong"'] & \mathcal{M} \otimes \mathcal{C} \arrow[dr, "\alpha"] & \\
& & \mathcal{C} \\
\mathcal{M} \otimes (\mathcal{M} \otimes \mathcal{C}) \arrow[r, "\mathcal{M}\otimes\alpha"'] & \mathcal{M} \otimes \mathcal{C} \arrow[ur, "\alpha"'] &
\end{tikzcd}$$

$$\begin{tikzcd}
\mathcal{C} \arrow[r, "\mathrm{id}"] \arrow[d, "\cong"'] & \mathcal{C} \\
1 \otimes \mathcal{C} \arrow[r, "\epsilon\otimes\mathcal{C}"'] & \mathcal{M} \otimes \mathcal{C} \arrow[u, "\alpha"']
\end{tikzcd}$$

Remark 3.2.13. An $\mathcal{M}$-module is also called an $\mathcal{M}$-action. We will sometimes say that $\mathcal{M}$ acts on $\mathcal{C}$.

Maps between $\mathcal{M}$-modules respecting the $\mathcal{M}$-action are called equivariant. More precisely:

Definition 3.2.14. An equivariant map between $\mathcal{M}$-modules $(\mathcal{C}, \alpha)$ and $(\mathcal{D}, \beta)$ is a map:

$$F \quad : \quad \mathcal{C} \to \mathcal{D} \tag{3.2.12}$$

such that the following square commutes:

$$\begin{tikzcd} \mathcal{M} \otimes \mathcal{C} \arrow[r, "\mathcal{M} \otimes F"] \arrow[d, "\alpha"'] & \mathcal{M} \otimes \mathcal{D} \arrow[d, "\beta"] \\ \mathcal{C} \arrow[r, "F"'] & \mathcal{D} \end{tikzcd}$$

There is a category of modules and equivariant maps.

Remark 3.2.15. The category of $\mathcal{M}$-module can be presented as the category of coalgebras for the comonad:

$$\mathcal{C} \quad \mapsto \quad \mathcal{M} \multimap \mathcal{C} \tag{3.2.13}$$

Its counit and comultiplication are induced by the unit and multiplication of $\mathcal{M}$. So a module is in some sense a *representable* comonad.

Remark 3.2.16. Considering a symmetric monoidal closed category $\mathcal{V}$ with all limits, we can define its categorical extensions by section and its categories of modules. Both notions are incomparable:

- A categorical extension by section is a category of coalgebras for a copointed endofunctor preserving all limits.
- A category of module is a category of coalgebra for a representable comonad.

So when comparing the two notions:

- Categories of module are more general in the sense that they can use their comultiplication to encode equality between composites of operations.
- Categorical extensions by section are more general in the sense that they can use any endofunctor preserving all limits, and not necessarily a representable one.

Remark 3.2.17. We can recast the categorical extension by section of Example 2.1.5 as a *representable* categorical extension by section, that is a categorical extension by section build from the endofunctor:

$$(\mathcal{G} \multimap \_) \quad : \quad \mathcal{V} \to \mathcal{V} \tag{3.2.14}$$

for $\mathcal{G} : \mathcal{V}$, copointed via a map:

$$V \quad : \quad \mathrm{Hom}_{\mathcal{V}}(1, \mathcal{G}) \tag{3.2.15}$$

Assuming that the free monoid $\mathcal{M}_{\mathcal{G}}$ generated by $\mathcal{G}$ with V as unit exists, giving a structure of coalgebra for this copointed endofunctor is equivalent to giving an $\mathcal{M}_{\mathcal{G}}$-module structure. In this case, such an extension is both an extension by a module structure and a categorical extension by section.

See Remark 2.1.8 for a definition of this free monoid as a colimit.

3.3. Induced and coinduced modules

Let $\mathcal{M}$ be a a notion of parametricity for $\mathcal{V}$, that is a monoid in $\mathcal{V}$. Now we prove that freely and cofreely $\mathcal{M}$-parametric objects always exist, and give compact descriptions for them. In categorical language, this means that there exist free and cofree modules over a (non-commutative) monoid $\mathcal{M}$ in a symmetric monoidal closed category $\mathcal{V}$. A proof can be found for example in [**Par77**] (Theorem 2.2 for free modules, Proposition 3.10 using $B = 1$ and $Q = A$ for cofree modules).

THEOREM 3.3.1. *The forgetful functor from $\mathcal{M}$-modules to $\mathcal{V}$ has both left and right adjoints. The left (resp. right) adjoint sends $\mathcal{C}$ to $\mathcal{M} \otimes \mathcal{C}$ (resp. $\mathcal{M} \multimap \mathcal{C}$) with the action of $\mathcal{M}$ induced by the canonical left (resp. right) action of $\mathcal{M}$ on itself.*

PROOF. We prove this result in linear simply-typed λ-calculus, so that it holds in any symmetric monoidal closed category. So it is crucial that any bound variable in our λ-terms occurs precisely once. An alternative direct proof by diagram chasing in $\mathcal{V}$ is of course possible.

We use the same notations as for simply-typed λ-calculus, for example for $c : \mathcal{C}$ and $d : \mathcal{D}$ we write:

$$(c, d) : \mathcal{C} \otimes \mathcal{D} \tag{3.3.1}$$

Moreover we denote the multiplication map of $\mathcal{M}$ by:

$$\_ \otimes \_ \quad : \quad \mathcal{M} \multimap \mathcal{M} \multimap \mathcal{M} \tag{3.3.2}$$

- We proceed with the proof for the right adjoint. Given $\mathcal{C} : \mathcal{V}$, we define $R(\mathcal{C})$ as $\mathcal{M} \multimap \mathcal{C}$ with the $\mathcal{M}$-action α defined by:

$$\alpha \quad : \quad \mathcal{M} \multimap R(\mathcal{C}) \multimap R(\mathcal{C}) \tag{3.3.3}$$
$$\alpha(i, u) \quad = \quad \lambda j.\, u(j \otimes i) \tag{3.3.4}$$

  Which is indeed an action. Now for $f : \mathcal{C} \to \mathcal{D}$ we define:

$$R(f) \quad : \quad R(\mathcal{C}) \multimap R(\mathcal{D}) \tag{3.3.5}$$
$$R(f, u) \quad = \quad f \circ u \tag{3.3.6}$$

  We see that $R(f)$ is equivariant, and that R is a functor from $\mathcal{V}$ to $\mathcal{M}$-modules.

  Now we want to check that R is right adjoint to the forgetful functor. Assume $(\mathcal{C}, \alpha)$ an $\mathcal{M}$-module, and $\mathcal{D} : \mathcal{V}$. We define:

$$\psi \quad : \quad \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{D}) \to \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{M} \multimap \mathcal{D}) \tag{3.3.7}$$
$$\psi(f) \quad = \quad \lambda c, i.\, f(\alpha(i, c)) \tag{3.3.8}$$

  and check that $\psi(f)$ is equivariant. Next we define:

$$\phi \quad : \quad \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{M} \multimap \mathcal{D}) \to \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{D}) \tag{3.3.9}$$
$$\phi(g) \quad = \quad \lambda c.\, g(c, 1) \tag{3.3.10}$$

  We check that for all $f : \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{D})$ we have:

$$\phi(\psi(f)) \quad = \quad \lambda c.\, (\lambda c', i.\, f(\alpha(i, c')))(c, 1) \tag{3.3.11}$$
$$= \quad \lambda c.\, f(\alpha(1, c)) \tag{3.3.12}$$
$$= \quad \lambda c.\, f(c) \tag{3.3.13}$$
$$= \quad f \tag{3.3.14}$$

  and that for all $g : \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{M} \multimap \mathcal{D})$ equivariant, meaning that:

$$g(\alpha(i, c)) \quad = \quad \lambda j.\, g(c, j \otimes i) \tag{3.3.15}$$

we have that:

$$\begin{aligned}
\psi(\phi(g)) &= \lambda c, i.\, (\lambda c'.\, g(c', 1))(\alpha(i, c)) && (3.3.16)\\
&= \lambda c, i.\, g(\alpha(i, c), 1) && (3.3.17)\\
&= \lambda c, i.\, (\lambda j.\, g(c, j \otimes i))(1) && (3.3.18)\\
&= \lambda c, i.\, g(c, 1 \otimes i) && (3.3.19)\\
&= \lambda c, i.\, g(c, i) && (3.3.20)\\
&= g && (3.3.21)
\end{aligned}$$

Now we just need naturality to conclude, so we check that for:

$$\begin{aligned}
\mathcal{C}, \mathcal{C}', \mathcal{D}, \mathcal{D}' &: \mathcal{V} && (3.3.22)\\
\alpha &: \mathcal{M} \text{ acting on } \mathcal{C} && (3.3.23)\\
\alpha' &: \mathcal{M} \text{ acting on } \mathcal{C}' && (3.3.24)\\
f &: \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}', \mathcal{C}) \text{ with } f \text{ equivariant} && (3.3.25)\\
g &: \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{D}) && (3.3.26)\\
h &: \mathrm{Hom}_{\mathcal{V}}(\mathcal{D}, \mathcal{D}') && (3.3.27)
\end{aligned}$$

we have that:

$$\begin{aligned}
\psi(h \circ g \circ f) &= \lambda c, i.\, h(g(f(\alpha'(i, c)))) && (3.3.28)\\
&= \lambda c, i.\, h(g(\alpha(i, f(c))) && (3.3.29)\\
&= \lambda c.\, R(h)(\lambda i.\, g(\alpha(i, f(c)))) && (3.3.30)\\
&= R(h) \circ \psi(g) \circ f && (3.3.31)
\end{aligned}$$

- Now we proceed with the left adjoint. For $\mathcal{C} : \mathcal{V}$ we define $L(\mathcal{C})$ as the module $\mathcal{M} \otimes \mathcal{C}$ with the action:

$$\begin{aligned}
\alpha &: \mathcal{M} \multimap L(\mathcal{C}) \multimap L(\mathcal{C}) && (3.3.32)\\
\alpha(i, (j, c)) &= (i \otimes j, c) && (3.3.33)
\end{aligned}$$

Given a map $f : \mathcal{C} \to \mathcal{D}$ we define:

$$\begin{aligned}
L(f) &: \mathcal{M} \otimes \mathcal{C} \to \mathcal{M} \otimes \mathcal{D} && (3.3.34)\\
L(f) &= \lambda(i, c).\, (i, f(c)) && (3.3.35)
\end{aligned}$$

We can check that $L(f)$ is equivariant, and that this gives a functor L from $\mathcal{V}$ to $\mathcal{M}$-modules.

Now we want to check that L is left adjoint to the forgetful functor. For $\mathcal{C} : \mathcal{V}$ and $(\mathcal{D}, \beta)$ an $\mathcal{M}$-module we define:

$$\begin{aligned}
\psi &: \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{D}) \to \mathrm{Hom}_{\mathcal{V}}(\mathcal{M} \otimes \mathcal{C}, \mathcal{D}) && (3.3.36)\\
\psi(f) &= \lambda(i, c).\, \beta(i, f(c)) && (3.3.37)
\end{aligned}$$

and we check that $\psi(f)$ is equivariant. Next we define:

$$\begin{aligned}
\phi &: \mathrm{Hom}_{\mathcal{V}}(\mathcal{M} \otimes \mathcal{C}, \mathcal{D}) \to \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{D}) && (3.3.38)\\
\phi(g) &= \lambda c.\, g(1, c) && (3.3.39)
\end{aligned}$$

We check that for all $f : \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{D})$ we have:

$$\begin{aligned} \phi(\psi(f)) &= \lambda c.\, \beta(1, f(c)) && (3.3.40)\\ &= \lambda c.\, f(c) && (3.3.41)\\ &= f && (3.3.42) \end{aligned}$$

and for all $g : \mathrm{Hom}_{\mathcal{V}}(\mathcal{M} \otimes \mathcal{C}, \mathcal{D})$ equivariant, meaning that:

$$\beta(i, g(j, c)) = g(i \otimes j, c) \qquad (3.3.43)$$

we have:

$$\begin{aligned} \psi(\phi(g)) &= \lambda(i, c).\, \beta(i, g(1, c)) && (3.3.44)\\ &= \lambda(i, c).\, g(i \otimes 1, c) && (3.3.45)\\ &= \lambda(i, c).\, g(i, c) && (3.3.46)\\ &= g && (3.3.47) \end{aligned}$$

Now we just need naturality to conclude, so we check that given:

$$\begin{aligned} \mathcal{C}, \mathcal{C}', \mathcal{D}, \mathcal{D}' &: \mathcal{V} && (3.3.48)\\ \beta &: \mathcal{M} \text{ acting on } \mathcal{D} && (3.3.49)\\ \beta' &: \mathcal{M} \text{ acting on } \mathcal{D}' && (3.3.50)\\ f &: \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}', \mathcal{C}) && (3.3.51)\\ g &: \mathrm{Hom}_{\mathcal{V}}(\mathcal{C}, \mathcal{D}) && (3.3.52)\\ h &: \mathrm{Hom}_{\mathcal{V}}(\mathcal{D}, \mathcal{D}') \text{ with } h \text{ equivariant} && (3.3.53) \end{aligned}$$

we have that:

$$\begin{aligned} \psi(h \circ g \circ f) &= \lambda(i, c).\, \beta'(i, h(g(f(c)))) && (3.3.54)\\ &= \lambda(i, c).\, h(\beta(i, g(f(c)))) && (3.3.55)\\ &= \lambda(i, c).\, h(\psi(g)(i, f(c))) && (3.3.56)\\ &= h \circ \psi(g) \circ L(f) && (3.3.57) \end{aligned}$$

□

Example 3.3.2. When we consider $\mathcal{V}$ the category of categories and $\square$ the category of semi-cubes:

- The cofreely $\square$-parametric category generated by $\mathcal{C}$ is the category $\mathcal{C}^{\square}$ of functors from $\square$ to $\mathcal{C}$, that is of semi-cubical objects in $\mathcal{C}$.
- The freely $\square$-parametric category generated by $\mathcal{C}$ is the category $\square \times \mathcal{C}$. The existence of this left adjoint is immediate, but this formula is pleasantly explicit.

It works the same for all the previously mentioned variants of cubes, including bicubes and augmented simplices.

Remark 3.3.3. Recall that we assumed that $\mathcal{V}$ was symmetric monoidal closed, as is the category of abelian group. We get the following correspondence:

Objects of $\mathcal{V}$	Models of type theory	Abelian groups
Monoids in $\mathcal{V}$	Notions of parametricity	Rings
Modules	Parametric models	Modules
Free modules	Freely parametric models	Induced modules
Cofree modules	Cubical models	Coinduced modules

3.4. Strict lex categories form a symmetric monoidal closed category

In order to apply the framework from Sections 3.2 and 3.3 to lex categories, we need to prove that they form a symmetric monoidal closed category. For this to work we have to use a variant of lex categories, which we call *strict lex categories*, such that:

- A strict lex category is a category with a chosen terminal object and chosen pullbacks.
- Functors between strict lex categories have to preserve the chosen limits on the nose.
- Limits in strict lex categories commute strictly, rather than up to natural isomorphisms.

These restrictions allow us to successfully use the 1-category of strict lex categories.

Remark 3.4.1. A better approach would be to use a monoidal closed 2-category of models of type theory, with a notion of parametricity defined as a monoid associative and unital up to 2-isomorphisms.

Remark 3.4.2. Most notions of parametricity considered here are finitely presented, so that assuming them strict is painless.

First we give our definition of strict lex categories. We will require that some morphisms are identities, implicitly requiring that their source and target are equal.

Definition 3.4.3. A strict lex category is a category $\mathcal{C}$ with:

- A terminal object $\top$.
- For any span:

$$\gamma \quad : \quad \Gamma \to \Delta \tag{3.4.1}$$
$$\theta \quad : \quad \Theta \to \Delta \tag{3.4.2}$$

  a pullback square:

$$\begin{tikzcd}
\Gamma \underset{\Delta}{\times} \Theta \arrow[r, "\pi_2"] \arrow[d, "\pi_1"'] & \Theta \arrow[d] \\
\Gamma \arrow[r] & \Delta
\end{tikzcd}$$

Moreover we ask that limits commute strictly, meaning that:

- The isomorphism:

$$\top \underset{\top}{\times} \top \quad \to \quad \top \tag{3.4.3}$$

  is an identity.
- Given a diagram:

$$\begin{tikzcd}
\Gamma_0 \arrow[r] \arrow[d] & \Delta_0 \arrow[d] & \Theta_0 \arrow[l] \arrow[d] \\
\Gamma_1 \arrow[r] & \Delta_1 & \Theta_1 \arrow[l] \\
\Gamma_2 \arrow[r] \arrow[u] & \Delta_2 \arrow[u] & \Theta_2 \arrow[l] \arrow[u]
\end{tikzcd}$$

the isomorphism from:

$$(\Gamma_0 \underset{\Gamma_1}{\times} \Gamma_2) \underset{\Delta_0 \underset{\Delta_1}{\times} \Delta_2}{\times} (\Theta_0 \underset{\Theta_1}{\times} \Theta_2) \tag{3.4.4}$$

to:

$$(\Gamma_0 \underset{\Delta_0}{\times} \Theta_0) \underset{\Gamma_1 \underset{\Delta_1}{\times} \Theta_1}{\times} (\Gamma_2 \underset{\Delta_2}{\times} \Theta_2) \tag{3.4.5}$$

is an identity.

Remark 3.4.4. The condition on pullbacks can be reformulated by saying that the limit of the three-by-three diagram can be computed row-by-row or column-by-column, yielding equal results.

We define morphisms of strict lex categories as functors preserving the chosen limits on the nose. They are called strict lex functors. This gives a category of strict lex categories. It is the category of algebras for an extension of the signature for categories, so that the functor forgetting limits has a left adjoint freely adding them.

Now we define the symmetric monoidal closed structure on the category of strict lex categories. The arrow is straightforward:

Definition 3.4.5. For $\mathcal{C}$ and $\mathcal{D}$ strict lex categories we define:

$$\mathcal{C} \multimap \mathcal{D} \tag{3.4.6}$$

as the strict lex category where:

- Objects are strict lex functors from $\mathcal{C}$ to $\mathcal{D}$.
- Morphisms are natural transformations.
- Limits are computed pointwise.

Remark 3.4.6. This definition would not be valid without the strict commutations of limits. Indeed assume given strict lex functors:

$$\alpha \quad : \quad F \to H \tag{3.4.7}$$

$$\beta \quad : \quad G \to H \tag{3.4.8}$$

We define their pullback pointwise, meaning that:

$$(F \underset{H}{\times} G)(\Gamma) \quad = \quad F(\Gamma) \underset{H(\Gamma)}{\times} G(\Gamma) \tag{3.4.9}$$

Then:

$$(F \underset{H}{\times} G)(\top) \quad = \quad \top \tag{3.4.10}$$

can only holds if:

$$\top \underset{\top}{\times} \top \quad = \quad \top \tag{3.4.11}$$

Similarly $F \underset{H}{\times} G$ commuting with a pullback:

$$\Gamma \underset{\Delta}{\times} \Theta \tag{3.4.12}$$

requires that the limit of:

$$\begin{tikzcd} F(\Gamma) \arrow[r] \arrow[d] & G(\Gamma) \arrow[d] & H(\Gamma) \arrow[l] \arrow[d] \\ F(\Delta) \arrow[r] & G(\Delta) & H(\Delta) \arrow[l] \\ F(\Theta) \arrow[r] \arrow[u] & G(\Theta) \arrow[u] & H(\Theta) \arrow[l] \arrow[u] \end{tikzcd}$$

computed row-by-row and column-by-column are equal.

Remark 3.4.7. The pointwise product of strict lex functors not being strict lex without the strict commutation of limits is analogous to the pointwise product of group morphisms not necessarily being a group morphism, when the target group is not abelian.

The tensor is defined so that it will be left adjoint to the arrow.

Definition 3.4.8. For $\mathcal{C}$ and $\mathcal{D}$ strict lex categories, we define:

$$\mathcal{C} \otimes \mathcal{D} \tag{3.4.13}$$

as the strict lex category freely generated by a functor:

$$\_ \otimes \_ \quad : \quad \mathcal{C} \times \mathcal{D} \to \mathcal{C} \otimes \mathcal{D} \tag{3.4.14}$$

such that the induced morphisms:

$$\begin{aligned} (\Gamma_0 \underset{\Gamma_1}{\times} \Gamma_2) \otimes \Delta &\quad\to\quad (\Gamma_0 \otimes \Delta) \underset{\Gamma_1 \otimes \Delta}{\times} (\Gamma_2 \otimes \Delta) && (3.4.15) \\ \Gamma \otimes (\Delta_0 \underset{\Delta_1}{\times} \Delta_2) &\quad\to\quad (\Gamma \otimes \Delta_0) \underset{\Gamma \otimes \Delta_1}{\times} (\Gamma \otimes \Delta_2) && (3.4.16) \\ \Gamma \otimes \top &\quad\to\quad \top && (3.4.17) \\ \top \otimes \Delta &\quad\to\quad \top && (3.4.18) \end{aligned}$$

are identities.

This means that in order to define a strict lex functor from $\mathcal{C} \otimes \mathcal{D}$ to $\mathcal{E}$, it is enough to define a functor:

$$F \quad : \quad \mathcal{C} \times \mathcal{D} \to \mathcal{E} \tag{3.4.19}$$

such that the induced morphisms:

$$\begin{aligned} F(\Gamma_0 \underset{\Gamma_1}{\times} \Gamma_2, \Delta) &\quad\to\quad F(\Gamma_0, \Delta) \underset{F(\Gamma_1, \Delta)}{\times} F(\Gamma_2, \Delta) && (3.4.20) \\ F(\Gamma, \Delta_0 \underset{\Delta_1}{\times} \Delta_2) &\quad\to\quad F(\Gamma, \Delta_0) \underset{F(\Gamma, \Delta_1)}{\times} F(\Gamma, \Delta_2) && (3.4.21) \\ F(\Gamma, \top) &\quad\to\quad \top && (3.4.22) \\ F(\top, \Delta) &\quad\to\quad \top && (3.4.23) \end{aligned}$$

are identities.

Remark 3.4.9. If we see strict lex categories as algebraic theories [**AR94**], then our tensor product of strict lex categories extends the tensor product of Lawvere theories [**Fre66**]. See [**HPP06**] for a computer-science oriented account on Lawvere theories and their tensor product.

Remark 3.4.10. The definition of our tensor implies that the expression:

$$(\Gamma_0 \underset{\Gamma_1}{\times} \Gamma_2) \otimes (\Delta_0 \underset{\Delta_1}{\times} \Delta_2) \tag{3.4.24}$$

can be distributed on the left or on the right, so that the limit of:

$$\begin{tikzcd}
\Gamma_0 \otimes \Delta_0 \arrow[r] \arrow[d] & \Gamma_1 \otimes \Delta_0 \arrow[d] & \Gamma_2 \otimes \Delta_0 \arrow[l] \arrow[d] \\
\Gamma_0 \otimes \Delta_1 \arrow[r] & \Gamma_1 \otimes \Delta_1 & \Gamma_2 \otimes \Delta_1 \arrow[l] \\
\Gamma_0 \otimes \Delta_2 \arrow[r] \arrow[u] & \Gamma_1 \otimes \Delta_2 \arrow[u] & \Gamma_2 \otimes \Delta_2 \arrow[l] \arrow[u]
\end{tikzcd}$$

computed row-by-row and column-by-column are equal. This equality would be unnatural, although not contradictory, without the strict commutation of limits.

Definition 3.4.11. We define 1 as the strict lex category freely generated by an object.

So giving a strict lex functor from 1 to $\mathcal{C}$ is the same as giving an object in $\mathcal{C}$.

Remark 3.4.12. The category with finite colimits freely generated by an object is the category of finite sets, so that 1 is equivalent to the opposite of the category of finite sets.

Remark 3.4.13. The monoidal structure on strict lex categories is similar to the one on abelian groups, with the following correspondence extending Remark 3.3.3.

Sets	Categories
Addition, zero	Finite limits
Abelian groups	Strict lex categories

We will make this formal in Remark 3.4.17.

We want to prove that Definitions 3.4.5, 3.4.8 and 3.4.11 give a symmetric monoidal closed structure on the category of strict lex categories. First we prove an auxiliary lemma. We denote by U the forgetful functor sending a strict lex category to its underlying category, and by L its left adjoint freely adding limits.

Lemma 3.4.14. *For any category $\mathcal{C}$ and strict lex category $\mathcal{D}$ we have a natural isomorphism:*

$$U(L(\mathcal{C}) \multimap \mathcal{D}) \quad \cong \quad U(\mathcal{D})^{\mathcal{C}} \tag{3.4.25}$$

Moreover limits in:

$$L(\mathcal{C}) \multimap \mathcal{D} \tag{3.4.26}$$

correspond to pointwise limits in $U(\mathcal{D})^{\mathcal{C}}$.

Proof. The isomorphism is immediate on objects. For any:

$$X \quad : \quad \mathrm{Hom}_{\mathrm{Lex}}(L(\mathcal{C}), \mathcal{D}) \tag{3.4.27}$$

we denote by $\overline{X}$ the corresponding functor from $\mathcal{C}$ to $U(\mathcal{D})$.

We denote by $\mathcal{D}^{\rightarrow}$ the arrow category of $\mathcal{D}$ with:

$$S, T \quad : \quad \mathcal{D}^{\rightarrow} \rightarrow \mathcal{D} \tag{3.4.28}$$

the functors giving the source and target of an arrow.

Assume given:

$$F, G \quad : \quad \mathrm{Hom}_{\mathrm{Lex}}(L(\mathcal{C}), \mathcal{D}) \tag{3.4.29}$$

Then a natural transformation from F to G is:

$$\{H : \mathrm{Hom}_{\mathrm{Cat}}(\mathcal{C}, \mathcal{D}^{\rightarrow}) \mid S \circ H = F,\ T \circ H = G\} \tag{3.4.30}$$

But using the fact that F and G are strict lex and the functoriality of H, we can show that any such H is in fact strict lex. Then we have:

$$\begin{aligned}
& \mathrm{Hom}_{L(\mathcal{C}) \multimap \mathcal{D}}(F, G) && (3.4.31)\\
\cong\ & \{H : \mathrm{Hom}_{\mathrm{Cat}}(L(\mathcal{C}), \mathcal{D}^{\rightarrow}) \mid S \circ H = F,\ T \circ H = G\} && (3.4.32)\\
\cong\ & \{H : \mathrm{Hom}_{\mathrm{Lex}}(L(\mathcal{C}), \mathcal{D}^{\rightarrow}) \mid S \circ H = F,\ T \circ H = G\} && (3.4.33)\\
\cong\ & \{\overline{H} : \mathrm{Hom}_{\mathrm{Cat}}(\mathcal{C}, U(\mathcal{D})^{\rightarrow}) \mid S \circ \overline{H} = \overline{F},\ T \circ \overline{H} = \overline{G}\} && (3.4.34)\\
\cong\ & \mathrm{Hom}_{U(\mathcal{D})^{\mathcal{C}}}(\overline{F}, \overline{G}) && (3.4.35)
\end{aligned}$$

This concludes the proof that:

$$U(L(\mathcal{C}) \multimap \mathcal{D}) \quad \cong \quad U(\mathcal{D})^{\mathcal{C}} \tag{3.4.36}$$

Moreover this isomorphism restricts elements in:

$$L(\mathcal{C}) \multimap \mathcal{D} \tag{3.4.37}$$

to $\mathcal{C}$, but as limits of such elements are computed pointwise, they correspond to pointwise limits in $U(\mathcal{D})^{\mathcal{C}}$. □

Remark 3.4.15. In principle a natural transformation between lex functors should be assumed lex. This condition can be omitted (and always is) because it holds for any natural transformation, as used in the proof above.

Now we bring all these constructions together for the main result of this section.

THEOREM 3.4.16. *The arrow, tensor and unit from Definitions 3.4.5, 3.4.8 and 3.4.11 give a symmetric monoidal closed structure on the category of strict lex categories.*

PROOF. There are many things to check:

- First we check that we have a natural isomorphism:

$$\mathrm{Hom}_{\mathrm{Lex}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{E}) \quad \cong \quad \mathrm{Hom}_{\mathrm{Lex}}(\mathcal{C}, \mathcal{D} \multimap \mathcal{E}) \tag{3.4.38}$$

  Indeed:

$$\mathrm{Hom}_{\mathrm{Lex}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{E}) \tag{3.4.39}$$

  is naturally equivalent to the set of:

$$F \quad : \quad \mathrm{Hom}_{\mathrm{Cat}}(\mathcal{C} \times \mathcal{D}, \mathcal{E}) \tag{3.4.40}$$

  such that the induced morphisms:

$$\begin{aligned}
F(\Gamma_0 \underset{\Gamma_1}{\times} \Gamma_2, \Delta) &\rightarrow F(\Gamma_0, \Delta) \underset{F(\Gamma_1, \Delta)}{\times} F(\Gamma_2, \Delta) && (3.4.41)\\
F(\top, \Delta) &\rightarrow \top && (3.4.42)\\
F(\Gamma, \Delta_0 \underset{\Delta_1}{\times} \Delta_2) &\rightarrow F(\Gamma, \Delta_0) \underset{F(\Gamma, \Delta_1)}{\times} F(\Gamma, \Delta_2) && (3.4.43)\\
F(\Gamma, \top) &\rightarrow \top && (3.4.44)
\end{aligned}$$

are identities, which is in turn naturally equivalent to the set of:

$$\overline{F} \quad : \quad \mathrm{Hom}_{\mathsf{Cat}}(\mathcal{C}, \mathcal{E}^{\mathcal{D}}) \tag{3.4.45}$$

Obeying the corresponding conditions.

- The fact that morphisms 3.4.41 and 3.4.42 are identities precisely means that $\overline{F}$ is a strict lex functor from $\mathcal{C}$ to $\mathcal{E}^{\mathcal{D}}$, with limits in $\mathcal{E}^{\mathcal{D}}$ computed pointwise.
- The fact that morphisms 3.4.43 and 3.4.44 are identity precisely means that the image of $\overline{F}$ is included in:

$$\mathcal{D} \multimap \mathcal{E} \tag{3.4.46}$$

which is the full subcategory of $\mathcal{E}^{\mathcal{D}}$ consisting of strict lex functors.

So together they precisely mean that:

$$\overline{F} \quad : \quad \mathrm{Hom}_{\mathsf{Lex}}(\mathcal{C}, \mathcal{E} \multimap \mathcal{D}) \tag{3.4.47}$$

- Next we check that the tensor product is symmetric. Indeed we can check that the the functor:

$$\mathrm{Sym} \quad : \quad \mathcal{C} \times \mathcal{D} \to \mathcal{D} \otimes \mathcal{C} \tag{3.4.48}$$

$$\mathrm{Sym}(c, d) \quad = \quad d \otimes c \tag{3.4.49}$$

commutes with limits in c and d, so that it can be extended to:

$$\mathrm{Sym} \quad : \quad \mathrm{Hom}_{\mathsf{Lex}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{D} \otimes \mathcal{C}) \tag{3.4.50}$$

We can check that Sym is self-inverse.

- Similarly we can define a functor:

$$\mathrm{Assoc} \quad : \quad (\mathcal{C} \times \mathcal{D}) \times \mathcal{E} \to \mathcal{C} \otimes (\mathcal{D} \otimes \mathcal{E}) \tag{3.4.51}$$

$$\mathrm{Assoc}((c, d), e) \quad = \quad c \otimes (d \otimes e) \tag{3.4.52}$$

which commutes with limits in each variable, so that it can be extended to:

$$\mathrm{Assoc} \quad : \quad \mathrm{Hom}_{\mathsf{Lex}}((\mathcal{C} \otimes \mathcal{D}) \otimes \mathcal{E}, \mathcal{C} \otimes (\mathcal{D} \otimes \mathcal{E})) \tag{3.4.53}$$

It is straightforward to define an inverse to Assoc.

- Now we need to check that 1 is indeed a unit. But this is a consequence of the natural isomorphism:

$$(1 \multimap \mathcal{C}) \quad \cong \quad \mathcal{C} \tag{3.4.54}$$

given by Lemma 3.4.14 applied to $1 = L(\top)$ with $\top$ the terminal category.

The checking of the various coherence diagrams is omitted. □

Remark 3.4.17. There exists a notion of commutative monad (see for example Section 6 in [**Bra14**]). For T a commutative monad on a symmetric monoidal closed category $\mathcal{C}$, the category of T-algebras is symmetric monoidal closed (assuming equalisers in $\mathcal{C}$ to build arrows and coequalisers in T-algebras to build tensors).

We give two examples.

- The monad for abelian groups on sets is commutative.
- The monad for strict lex categories on categories is commutative. The monad for lex categories is only commutative in a 2-categorical sense.

So the monoidal structure on abelian groups and strict lex categories can both be built this way, cementing the analogy from Remark 3.4.13.

Remark 3.4.18. If we uses a suitable 2-category of lex categories instead of strictifying, we could prove that it is a *pseudo-closed* 2-category in two steps:

- The 2-monad for lex categories is pseudo-commutative by [**Fra11**].
- Pseudo-commutative 2-monads have pseudo-closed 2-categories of algebras by [**HP02**].

3.5. Notions of parametricity for strict lex categories

Now we can use the symmetric monoidal closed structure from the previous section to define notions of parametricity and parametric models. We emphasis this:

Definition 3.5.1. A notion of parametricity for strict lex categories is a monoid in the category of strict lex categories.

We unfold this definition:

Proposition 3.5.2. *Giving a notion of parametricity for strict lex categories is equivalent to giving a strict lex category $\mathcal{M}$ with a (strictly) monoidal product $\_ \otimes \_$ such that the canonical morphisms:*

$$(\Gamma_0 \underset{\Gamma_1}{\times} \Gamma_2) \otimes \Gamma \quad\to\quad (\Gamma_0 \otimes \Gamma) \underset{\Gamma_1 \otimes \Gamma}{\times} (\Gamma_2 \otimes \Gamma) \tag{3.5.1}$$

$$\top \otimes \Gamma \quad\to\quad \top \tag{3.5.2}$$

$$\Gamma \otimes (\Gamma_0 \underset{\Gamma_1}{\times} \Gamma_2) \quad\to\quad (\Gamma \otimes \Gamma_0) \underset{\Gamma \otimes \Gamma_1}{\times} (\Gamma \otimes \Gamma_2) \tag{3.5.3}$$

$$\Gamma \otimes \top \quad\to\quad \top \tag{3.5.4}$$

are identities.

We will call such a category a monoidal strict lex category. We prove that notions of parametricity for categories can be extended to strict lex categories.

Proposition 3.5.3. *The functor L freely adding finite limits to a category is strongly monoidal, meaning that we have natural isomorphisms:*

$$L(\mathcal{C} \times \mathcal{D}) \quad\cong\quad L(\mathcal{C}) \otimes L(\mathcal{D}) \tag{3.5.5}$$

$$L(\top) \quad\cong\quad 1 \tag{3.5.6}$$

obeying some coherence conditions.

Proof. We write U for the functor forgetting finite limits.

- We have a string of natural isomorphisms where $\mathcal{C}$ and $\mathcal{D}$ are categories and $\mathcal{E}$ is a strict lex category:

$$\mathrm{Hom}_{\mathrm{Lex}}(L(\mathcal{C} \times \mathcal{D}), \mathcal{E}) \quad\cong\quad \mathrm{Hom}_{\mathrm{Cat}}(\mathcal{C} \times \mathcal{D}, U(\mathcal{E})) \tag{3.5.7}$$

$$\cong\quad \mathrm{Hom}_{\mathrm{Cat}}(\mathcal{C}, U(\mathcal{E})^{\mathcal{D}}) \tag{3.5.8}$$

$$\cong\quad \mathrm{Hom}_{\mathrm{Cat}}(\mathcal{C}, U(L(\mathcal{D}) \multimap \mathcal{E})) \tag{3.5.9}$$

$$\cong\quad \mathrm{Hom}_{\mathrm{Lex}}(L(\mathcal{C}), L(\mathcal{D}) \multimap \mathcal{E}) \tag{3.5.10}$$

$$\cong\quad \mathrm{Hom}_{\mathrm{Lex}}(L(\mathcal{C}) \otimes L(\mathcal{D}), \mathcal{E}) \tag{3.5.11}$$

  where Equation 3.5.9 uses Lemma 3.4.14. We can conclude by Yoneda lemma.
- The isomorphism between $L(\top)$ and 1 with $\top$ the terminal category is an immediate consequence of the definition of 1.

The checking of the various coherence diagrams is omitted. □

Corollary 3.5.4. *For any notion of parametricity for categories $\mathcal{M}$, we have that $L(\mathcal{M})$ is a notion of parametricity for strict lex categories.*

Proof. Strongly monoidal functors preserve monoids. □

3.6. Non-iterated parametricity and truncated cubes

Using lex categories, we can present truncated notions of parametricity, i.e. non-iterated ones. We will use *non-strict* lex categories, see Remark 3.6.8 for a discussion.

For the rest of this section, let $\square$ be the category of semi-cube or the category of cubes with reflexivities only, and let n be fixed a natural number.

Definition 3.6.1. Any object in $\square$ if of the product of k copies of $\mathbb{I}$ for some k, written as:

$$\mathbb{I}^k = \mathbb{I} \otimes \cdots \otimes \mathbb{I} \tag{3.6.1}$$

We write $\square_n$ the full subcategory of $\square$ with objects:

$$1, \mathbb{I}, \cdots, \mathbb{I}^n \tag{3.6.2}$$

Our goal is to show that $\square_n$ should induce an n-truncated notion of parametricity for lex categories. More precisely we want to check that the free lex category generated by $\square_n$ is a monoidal lex category.

Recall that the Day convolution extends the monoidal tensor on $\square$ to a monoidal tensor on $\mathrm{Set}^{\square}$. This tensor is closed on both sides, meaning that we have two functors:

$$\begin{aligned} \_ \multimap \_ \quad &: \quad \mathrm{Set}^{\square} \to \mathrm{Set}^{\square} \to \mathrm{Set}^{\square} &(3.6.3)\\ \_ \multimapinv \_ \quad &: \quad \mathrm{Set}^{\square} \to \mathrm{Set}^{\square} \to \mathrm{Set}^{\square} &(3.6.4)\end{aligned}$$

with natural isomorphisms:

$$\begin{aligned} \mathrm{Hom}(X \otimes Y, Z) \quad &\cong \quad \mathrm{Hom}(X, Y \multimap Z) &(3.6.5)\\ \mathrm{Hom}(X \otimes Y, Z) \quad &\cong \quad \mathrm{Hom}(Y, Z \multimapinv X) &(3.6.6)\end{aligned}$$

for X, Y and Z in $\mathrm{Set}^{\square}$.

The inclusion of full subcategory:

$$f \quad : \quad \square_n \to \square \tag{3.6.7}$$

induces a post-composition functor:

$$f^* \quad : \quad \mathrm{Set}^{\square} \to \mathrm{Set}^{\square_n} \tag{3.6.8}$$

with a full and faithful left (resp. right) adjoint $f_!$ (resp. f_*). An object $X : \mathrm{Set}^{\square}$ is called coskeletal if:

$$X \quad \cong \quad f_*(f^*(X)) \tag{3.6.9}$$

We have that $f_*(f^*(X))$ is always coskeletal. It is called the coskeleton of X. We want to give a helpful criteria for coskeletal object. First we define n-cells inductively:

Definition 3.6.2. We define:

$$\delta\mathbb{I}^k \quad : \quad \mathrm{Set}^{\square} \tag{3.6.10}$$
$$\delta i^k \quad : \quad \delta\mathbb{I}^k \to \mathbb{I}^k \tag{3.6.11}$$

inductively on n by:

$$\delta\mathbb{I}^0 \quad = \quad \perp \tag{3.6.12}$$
$$\delta i^0 \quad = \quad \eta_1 \tag{3.6.13}$$

where η_1 is the unique morphism from $\perp$ to 1, and by building δi^{k+1} from δi^k using the pushout square:

$$\begin{tikzcd}
 & & \delta\mathbb{I}^k \coprod \delta\mathbb{I}^k \arrow[dll, "\alpha_{\delta\mathbb{I}^k}"'] \arrow[drr, "\delta i^k \coprod \delta i^k"] & & \\
\delta\mathbb{I}^k \otimes \mathbb{I} \arrow[drr] \arrow[ddrr, bend right, "\delta i^k \otimes \mathbb{I}"'] & & & & \mathbb{I}^n \coprod \mathbb{I}^n \arrow[dll] \arrow[ddll, bend left, "\alpha_{\mathbb{I}^k}"] \\
 & & \delta\mathbb{I}^{k+1} \arrow[d, "\delta i^{k+1}"] & & \\
 & & \mathbb{I}^{k+1} & &
\end{tikzcd}$$

where:

$$\alpha_X = (X \otimes d^0 \mid X \otimes d^1): \quad : \quad X \coprod X \to X \otimes \mathbb{I} \tag{3.6.14}$$

It is possible to check that the map:

$$\delta i^k \quad : \quad \delta\mathbb{I}^k \to \mathbb{I}^k \tag{3.6.15}$$

is the inclusion of the border of a k-cube in the usual sense.

Definition 3.6.3. A morphism:

$$u \quad : \quad A \to B \tag{3.6.16}$$

is called left orthogonal to an object X if the induced map:

$$u^* \quad : \quad \mathrm{Hom}(B, X) \to \mathrm{Hom}(A, X) \tag{3.6.17}$$

is a bijection. We denote this by $u \perp X$.

So u is left orthogonal to X if for any:

$$f \quad : \quad A \to X \tag{3.6.18}$$

there exists a unique dotted arrow making the following triangle commutes:

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "u"'] & X \\
B \arrow[ur, dashed] &
\end{tikzcd}$$

We do not prove the next lemma. It holds both for semi-cubes and cubes with refexivities (as claimed without proof in [**KRRZ11**]), although we do not know to what extent it holds for other cubes.

Lemma 3.6.4. *An element $X : \mathrm{Set}^{\square}$ is coskeletal if and only if for all $k > n$ we have:*

$$\delta i^k \perp X \tag{3.6.19}$$

Lemma 3.6.5. *If $X : \mathrm{Set}^{\square}$ is coskeletal, then so are:*

$$\mathbb{I} \multimap X \tag{3.6.20}$$

and:

$$X \multimapinv \mathbb{I} \tag{3.6.21}$$

Proof. We prove the assertion on $\mathbb{I} \multimap X$.

- We need to prove that:
$$\delta i^k \perp (\mathbb{I} \multimap X) \tag{3.6.22}$$
for all $k > n$. This is equivalent to:
$$(\delta i^k \otimes \mathbb{I}) \perp X \tag{3.6.23}$$
We can decompose $\delta i^k \otimes \mathbb{I}$ as:
$$\delta \mathbb{I}^k \otimes \mathbb{I} \longrightarrow \delta \mathbb{I}^{k+1} \xrightarrow{\delta i^{k+1}} \mathbb{I}^{k+1}$$
where the first map is a pushout of $\delta i^k \coprod \delta i^k$. But we can conclude since maps left orthogonal to X are stable under coproducts, pushouts and composition.

To prove the assertion on $X \multimapinv \mathbb{I}$, we need to rework this whole section in mirror, using an alternative equivalent definition of $\delta \mathbb{I}^{k+1}$ based on $\mathbb{I} \otimes \delta \mathbb{I}^k$ rather than $\delta \mathbb{I}^k \otimes \mathbb{I}$. ◻

Now we are ready to restrict the Day convolution product from $\mathrm{Set}^{\square}$ to $\mathrm{Set}^{\square_n}$.

Lemma 3.6.6. *The category $\mathrm{Set}^{\square_n}$ inherits a monoidal structure from the Day convolution on $\mathrm{Set}^{\square}$. This induced tensor on $\mathrm{Set}^{\square_n}$ commutes with colimits in both variables.*

Proof. We proceed in three steps.

- If $Y : \mathrm{Set}^{\square}$ is coskeletal, so are:
$$\mathbb{I} \multimap Y \tag{3.6.24}$$
and:
$$Y \multimapinv \mathbb{I} \tag{3.6.25}$$
This is Lemma 3.6.5, and this is the only part which rely on the cube category $\square$.
- If $Y : \mathrm{Set}^{\square}$ is coskeletal, so are
$$X \multimap Y \tag{3.6.26}$$
and:
$$Y \multimapinv X \tag{3.6.27}$$
for any $X : \mathrm{Set}^{\square}$. Indeed:
  - By iterating the previous point, we know that the property holds when $X = \mathbb{I}^k$, that is when X is representable.

- Since coskeletal objects are stable under limits (as f_* and f^* preserve limits), the property is stable under colimits.
- We can conclude because any X is a colimit of representables.

- For any $X, Y : \mathrm{Set}^{\square}$ we have natural isomorphisms:

$$\begin{aligned} f^*(f_! f^*(X) \otimes Y) &\cong f^*(X \otimes Y) && (3.6.28)\\ f^*(X \otimes f_! f^*(Y)) &\cong f^*(X \otimes Y) && (3.6.29) \end{aligned}$$

For example we can prove the first isomorphism using Yoneda lemma and the following string of natural isomorphisms:

$$\begin{aligned} \mathrm{Hom}(f^*(f_! f^*(X) \otimes Y), Z) &\cong \mathrm{Hom}(f_! f^*(X) \otimes Y, f_*(Z)) && (3.6.30)\\ &\cong \mathrm{Hom}(f_! f^*(X), Y \multimap f_*(Z)) && (3.6.31)\\ &\cong \mathrm{Hom}(X, f_* f^*(Y \multimap f_*(Z))) && (3.6.32)\\ &\cong \mathrm{Hom}(X, Y \multimap f_*(Z)) && (3.6.33)\\ &\cong \mathrm{Hom}(X \otimes Y, f_*(Z)) && (3.6.34)\\ &\cong \mathrm{Hom}(f^*(X \otimes Y), Z) && (3.6.35) \end{aligned}$$

where Equation 3.6.33 used the fact that $f_*(Z)$ is coskeletal, so that:

$$Y \multimap f_*(Z) \tag{3.6.36}$$

is coskeletal as well by the previous point.

- For $X, Y : \mathrm{Set}^{\square_n}$, we define:

$$\begin{aligned} X \otimes_n Y &= f^*(f_!(X) \otimes f_!(Y)) && (3.6.37)\\ 1_n &= f^*(1) && (3.6.38) \end{aligned}$$

Using the previous point we can check that this gives a monoidal structure.

The functor $\_ \otimes_n \_$ commutes with colimits in both variables because so does $\_ \otimes \_$, and f^* and $f_!$ commute with colimits. □

The proof rely on the fact that if Y is coskeletal, so are $\mathbb{I} \multimap Y$ and $Y \mathbin{\circ\!\!-} \mathbb{I}$. It holds for semi-cubes and cubes with reflexivities, but we do not know whether it holds for other cubes.

Proposition 3.6.7. *The free lex category generated by $\square_n$ is (non-strict) monoidal.*

Proof. By duality, the previous lemma gives a monoidal structure $\_ \otimes_n \_$ on $(\mathrm{Set}^{\square_n})^{op}$, commuting with limits in both variables.

The free lex category generated by $\square_n$ is equivalent to the closure of representables in $(\mathrm{Set}^{\square_n})^{op}$ under finite limits. We want to restrict $\_ \otimes_n \_$ to this closure. So we want to prove that for X and Y finite colimits of representables in $\mathrm{Set}^{\square_n}$, we have that:

$$X \otimes_n Y \tag{3.6.39}$$

is a finite colimit of representables. As $\_ \otimes_n \_$ commutes with colimits in each variable, it is sufficient to prove this for X and Y representable. This means that we have to prove that:

$$\mathbb{I}^k \otimes_n \mathbb{I}^{k'} = f^*(\mathbb{I}^k \otimes \mathbb{I}^{k'}) \tag{3.6.40}$$

is a finite colimit of representables for $k, k' \leq n$ in order to conclude.

We will prove the more general fact that $f^*(\mathbb{I}^l)$ is a finite colimit of representables in $\mathrm{Set}^{\square_n}$ for all l.

We know that for any object $X : \mathrm{Set}^{\square_n}$ we have:

$$X \quad \cong \quad \mathrm{colim}_{(i:\square_n^{op})\times X(i)} \mathrm{Hom}_{\square}(i, \_) \tag{3.6.41}$$

so that X is a colimit of representable. This is called the co-Yoneda lemma.

If $X = f^*(\mathbb{I}^l)$, then $X(i)$ is finite (as $X(i) = \mathrm{Hom}_{\square}(l, i)$ and $\square$ is locally finite) and $\square_n$ is finite, so that X is a finite colimit of representables. $\square$

We crucially used the fact that $\square$ is locally finite (i.e. has finite sets of morphisms).

Remark 3.6.8. The monoidal category from the previous proposition is not strict, so that it is not technically a notion of parametricity. We expect that this could be worked around using a 2-category of lex categories, or alternatively a strictification result for lex monoidal categories.

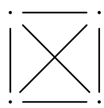
A $\square_n$-parametric lex category should have an endofunctor $\_{*}$ such that for any X, the object $X_{*(n+1)}$ can be computed as a limit of copies of X_{*k} with $k \leq n$. A cofreely $\square_n$-parametric lex category should simply be a category of n-truncated cubical objects in some lex category.

Remark 3.6.9. Type-theoretically, the condition on $\_{*}$ can be reformulated as:

$$X_{*(n+1)} \quad = \quad \top \tag{3.6.42}$$

Indeed the limit is represented by the complicated context in which $X_{*(n+1)}$ is defined.

Remark 3.6.10. The final result might hold even for cubes with diagonals where Lemma 3.6.4 fails, using another inductive definition for δi^k where the border of a square is:



Remark 3.6.11. This examples of n-truncated parametricity cannot be formulated as a categorical extension by section for $n > 0$. Indeed we add operations $\_{*}$ with an equation on $X_{*(n+1)}$. But consider the much simpler case where we extend $X : \mathcal{U}$ by:

$$s \quad : \quad X \to X \tag{3.6.43}$$
$$\_ \quad : \quad (x : X) \to s(s(x)) = x \tag{3.6.44}$$

This cannot give a categorical extension by section (i.e. be a coalgebra for a copointed endofunctors), as in order consider a composition of operations such as $s \circ s$ we need to use a comonad and not a copointed endofunctor. Similarly we cannot add equations on $X_{*(n+1)}$ without a comonad structure.

3.7. Clans with strictly commuting limits

As for lex categories in Section 3.4, we need to assume some strictness conditions on clans for them to form a symmetric monoidal closed category. These clans with strictly commuting limits will be called *strict clans*. Recall the definition of clan:

Definition 3.7.1. A clan is a category with a terminal object $\top$, together with a class of maps called fibrations such that:

- Fibrations are stable under isomorphisms, composition and pullbacks.
- Maps to $\top$ are fibrations.

We give some vocabulary:

Definition 3.7.2. A Reedy fibrant square in a clan is a square of fibrations:

$$\begin{tikzcd} \Delta \arrow[r, two heads] \arrow[d, two heads] & B \arrow[d, two heads] \\ A \arrow[r, two heads] & \Gamma \end{tikzcd}$$

such that we have an induced fibration:

$$\Delta \quad \twoheadrightarrow \quad A \underset{\Gamma}{\times} B \tag{3.7.1}$$

Remark 3.7.3. Type-theoretically, a Reedy fibrant square corresponds to a context Γ with:

$$\Gamma \quad \vdash \quad A \tag{3.7.2}$$

$$\Gamma \quad \vdash \quad B \tag{3.7.3}$$

$$\Gamma, A, B \quad \vdash \quad C \tag{3.7.4}$$

Indeed from such types we can build a Reedy fibrant square:

$$\begin{tikzcd} \Gamma, A, B, C \arrow[r, "\mathrm{w}_{A,C}"] \arrow[d, "\mathrm{w}_{B,C}"'] & \Gamma, B \arrow[d, "\mathrm{w}_B"] \\ \Gamma, A \arrow[r, "\mathrm{w}_A"'] & \Gamma \end{tikzcd}$$

We are ready to define strict clans.

Definition 3.7.4. A strict clan is a clan with strictly commuting limits. This means that:

- The canonical morphism:

$$\top \underset{\top}{\times} \top \to \top \tag{3.7.5}$$

  is an identity.
- Given a diagram:

$$\begin{tikzcd} \Gamma_0 \arrow[r, two heads] \arrow[d, two heads] & \Delta_0 \arrow[d, two heads] & \Theta_0 \arrow[l] \arrow[d, two heads] \\ \Gamma_1 \arrow[r, two heads] & \Delta_1 & \Theta_1 \arrow[l] \\ \Gamma_2 \arrow[r, two heads] \arrow[u] & \Delta_2 \arrow[u] & \Theta_2 \arrow[l] \arrow[u] \end{tikzcd}$$

  where the top left square is Reedy fibrant, the isomorphism from:

$$(\Gamma_0 \underset{\Gamma_1}{\times} \Gamma_2) \underset{\Delta_0 \underset{\Delta_1}{\times} \Delta_2}{\times} (\Theta_0 \underset{\Theta_1}{\times} \Theta_2) \tag{3.7.6}$$

to:

$$(\Gamma_0 \underset{\Delta_0}{\times} \Theta_0) \underset{\Gamma_1 \underset{\Delta_1}{\times} \Theta_1}{\times} (\Gamma_2 \underset{\Delta_2}{\times} \Theta_2) \tag{3.7.7}$$

is an identity.

CHECK. This to be well-defined because for any diagram:

$$\begin{tikzcd} A \arrow[r, two heads] \arrow[d] & \Gamma \arrow[d] & \Delta \arrow[l] \arrow[d] \\ A' \arrow[r, two heads] & \Gamma' & \Delta' \arrow[l] \end{tikzcd}$$

with the left square Reedy fibrant, we have an induced fibration:

$$A \underset{\Gamma}{\times} \Delta \quad \twoheadrightarrow \quad A' \underset{\Gamma'}{\times} \Delta' \tag{3.7.8}$$

Indeed this induced map is isomorphic to the composite of pullbacks of fibrations as follows:

$$\begin{tikzcd} A \underset{\Gamma}{\times} \Delta \arrow[r] \arrow[d] & A \arrow[d] \\ (A' \underset{\Gamma'}{\times} \Gamma) \underset{\Gamma}{\times} \Delta \arrow[r] \arrow[d, "\cong"'] & (A' \underset{\Gamma'}{\times} \Gamma) \\ A' \underset{\Gamma'}{\times} \Delta \arrow[r] \arrow[d] & \Delta \arrow[d] \\ A' \underset{\Gamma'}{\times} \Delta' \arrow[r] & \Delta' \end{tikzcd}$$

□

Morphisms between strict clans are defined as functors preserving fibrations and commuting with all constructors up to equality. So the category of strict clans is a category of algebras for a signature.

We give an explicit description for fibrations in a strict clan freely generated by a category:

Lemma 3.7.5. *Fibrations in the strict clan freely generated by a category are precisely maps isomorphic to projections.*

PROOF. It is straightforward to see that any clan has cartesian products, and that any map isomorphic to a cartesian projection is a fibration. Now we prove that the class of maps isomorphic to a projection is a valid class of fibrations:

- They are stable under isomorphisms by definition.
- They are stable under composition as for any objects Γ, Δ and Θ we have:

$$\begin{tikzcd} (\Gamma \times \Delta) \times \Theta \arrow[r, "\pi_1"] \arrow[d, "\cong"'] & \Gamma \times \Delta \arrow[r, "\pi_1"] & \Gamma \arrow[d, "\cong"] \\ \Gamma \times (\Delta \times \Theta) \arrow[rr, "\pi_1"'] & & \Gamma \end{tikzcd}$$

so a composite of projections is isomorphic to a projection.

- They are stable under pullbacks because for any objects Γ, Δ, Θ with:

$$\sigma \quad : \quad \Theta \to \Gamma \tag{3.7.9}$$

  we have a pullback square:

$$\begin{tikzcd} \Theta \times \Delta \arrow[r, "\sigma \times \Delta"] \arrow[d, "\pi_1"'] & \Gamma \times \Delta \arrow[d, "\pi_1"] \\ \Theta \arrow[r, "\sigma"'] & \Gamma \end{tikzcd}$$

  so the pullback of a projection is isomorphic to a projection.
- Maps to $\top$ are isomorphic to projections:

$$\Gamma \xrightarrow{\cong} \top \times \Gamma \xrightarrow{\pi_1} \top$$

So fibrations are precisely maps isomorphic to projections. ☐

Remark 3.7.6. While the clan freely generated by a category $\mathcal{C}$ is the cartesian category freely generated by $\mathcal{C}$, it is not clear to us what precisely is the *strict* clan freely generated by $\mathcal{C}$.

3.8. Strict clans form a symmetric monoidal closed category

Now we define the arrow, tensor and unit for strict clans.

Definition 3.8.1. Let $\mathcal{C}$ and $\mathcal{D}$ be strict clans. We define:

$$\mathcal{C} \multimap \mathcal{D} \tag{3.8.1}$$

as the strict clan where:

- Objects are morphisms of strict clans from $\mathcal{C}$ to $\mathcal{D}$.
- Morphisms are natural transformations between the underlying functors.
- Limits are computed pointwise.
- A natural transformation:

$$\alpha \quad : \quad F \to G \tag{3.8.2}$$

  is a fibration if for any fibration:

$$A \twoheadrightarrow \Gamma \tag{3.8.3}$$

  in $\mathcal{C}$ we have an induced fibration:

$$F(A) \twoheadrightarrow G(A) \underset{G(\Gamma)}{\times} F(\Gamma) \tag{3.8.4}$$

Check. First we prove that fibrations are in particular pointwise fibrations:

- For a fibration:

$$\alpha \quad : \quad F \twoheadrightarrow G \tag{3.8.5}$$

  and Γ in $\mathcal{C}$, from the fibration:

$$\Gamma \twoheadrightarrow \top \tag{3.8.6}$$

  we get an induced fibration:

$$H(\Gamma) \twoheadrightarrow G(\Gamma) \underset{G(\top)}{\times} H(\top) \tag{3.8.7}$$

  but this map is isomorphic to:

$$\alpha_\Gamma \quad : \quad H(\Gamma) \to G(\Gamma) \tag{3.8.8}$$

Now we want to check that:

$$\mathcal{C} \multimap \mathcal{D} \tag{3.8.9}$$

is indeed a strict clan. First we check that limits are well-defined:

- We check that the constant functor $F_\top$ with value $\top$ is a morphism of strict clan, giving a terminal object in:

  $$\mathcal{C} \multimap \mathcal{D} \tag{3.8.10}$$

  Indeed the induced map:

  $$F_\top(A \underset{\Gamma}{\times} \Delta) \quad \to \quad F_\top(A) \underset{F_\top(\Gamma)}{\times} F_\top(\Delta) \tag{3.8.11}$$

  is an identity because it is equal to the unique map in:

  $$\top \quad \to \quad \top \underset{\top}{\times} \top \tag{3.8.12}$$

  whose inverse is assumed to be an identity.
- Now we check that pullbacks can be defined pointwise in:

  $$\mathcal{C} \multimap \mathcal{D} \tag{3.8.13}$$

  So we assume given a diagram:

  $$\begin{tikzcd} F \arrow[r, two heads] & G & H \arrow[l] \end{tikzcd}$$

  and we define $F \underset{G}{\times} H$ by:

  $$(F \underset{G}{\times} H)(\Gamma) \quad = \quad F(\Gamma) \underset{G(\Gamma)}{\times} H(\Gamma) \tag{3.8.14}$$

  This is well defined as we have a fibration:

  $$F(\Gamma) \twoheadrightarrow G(\Gamma) \tag{3.8.15}$$

  We check that $F \underset{G}{\times} H$ commutes with limits.
  - We need to check that the morphism:

    $$(F \underset{G}{\times} H)(\top) \quad \to \quad \top \tag{3.8.16}$$

    is an identity. But this map is equal to the unique map in:

    $$\top \underset{\top}{\times} \top \quad \to \quad \top \tag{3.8.17}$$

    which is assumed to be an identity.
  - To check that $F \underset{G}{\times} H$ commutes with pullbacks, we consider a diagram:

    $$\begin{tikzcd} A \arrow[r, two heads] & \Gamma & \Delta \arrow[l] \end{tikzcd}$$

    in $\mathcal{C}$. We need to prove that the isomorphism between the limit of the diagram:

    $$\begin{tikzcd} F(A) \arrow[r, two heads] \arrow[d, two heads] & F(\Gamma) \arrow[d, two heads] & F(\Delta) \arrow[l] \arrow[d, two heads] \\ G(A) \arrow[r, two heads] & G(\Gamma) & G(c) \arrow[l] \\ H(A) \arrow[r, two heads] \arrow[u] & H(\Gamma) \arrow[u] & H(\Delta) \arrow[l] \arrow[u] \end{tikzcd}$$

computed row-by-row and column-by-column is an identity. This holds by the strict commutation of limits, because the top left square is Reedy fibrant by definition of fibrations in:

$$\mathcal{C} \multimap \mathcal{D} \tag{3.8.18}$$

Next we check that the class of fibrations is suitably closed.

- Stability under isomorphisms is straightforward.
- Composition preserves fibrations. Indeed for a diagram:

$$F \twoheadrightarrow G \twoheadrightarrow H$$

we need to prove that for any fibration:

$$A \twoheadrightarrow \Gamma \tag{3.8.19}$$

in $\mathcal{C}$ the induced map:

$$F(A) \quad \to \quad H(A) \underset{H(\Gamma)}{\times} F(\Gamma) \tag{3.8.20}$$

is a fibration in $\mathcal{D}$. But this map is isomorphic to the composite of a fibration and a pullback of fibration as follows:

$$\begin{tikzcd}
F(A) \arrow[d, two heads] & \\
G(A) \underset{G(\Gamma)}{\times} F(\Gamma) \arrow[r] \arrow[d, two heads] & G(A) \arrow[d, two heads] \\
H(A) \underset{H(\Gamma)}{\times} G(\Gamma) \underset{G(\Gamma)}{\times} F(\Gamma) \arrow[r] \arrow[d, "\cong"] & H(A) \underset{H(\Gamma)}{\times} G(\Gamma) \\
H(A) \underset{H(\Gamma)}{\times} F(\Gamma) &
\end{tikzcd}$$

- Pointwise pullbacks of fibrations are fibrations. We need to check that given a diagram:

$$F \twoheadrightarrow G \longleftarrow H$$

in $\mathcal{C} \multimap \mathcal{D}$, the projection:

$$F \underset{G}{\times} H \to H \tag{3.8.21}$$

is a fibration. So we need to prove that for any fibration:

$$A \twoheadrightarrow \Gamma \tag{3.8.22}$$

in $\mathcal{C}$, the induced map:

$$(F \underset{G}{\times} H)(A) \quad \to \quad H(A) \underset{H(\Gamma)}{\times} (F \underset{G}{\times} H)(\Gamma) \tag{3.8.23}$$

is a fibration. But this map is isomorphic to the map:

$$F(A) \underset{G(A)}{\times} H(A) \quad \to \quad F(\Gamma) \underset{G(\Gamma)}{\times} H(A) \tag{3.8.24}$$

which is isomorphic to a pullback of fibration as follows:

$$\begin{tikzcd}
F(A) \underset{G(A)}{\times} H(A) \arrow[r] \arrow[d] & F(A) \arrow[d, two heads] \\
F(\Gamma) \underset{G(\Gamma)}{\times} G(A) \underset{G(A)}{\times} H(A \arrow[r] \arrow[d, "\cong"] & F(\Gamma) \underset{G(\Gamma)}{\times} G(A) \\
F(\Gamma) \underset{G(\Gamma)}{\times} H(A) &
\end{tikzcd}$$

- The unique natural transformation from any morphism F to the functor with constant value $\top$ is a fibration. Indeed for any fibration:

  $$A \twoheadrightarrow \Gamma \tag{3.8.25}$$

  we need to check that the induced map:

  $$F(A) \quad \to \quad \top \underset{\top}{\times} F(\Gamma) \tag{3.8.26}$$

  is a fibration in $\mathcal{D}$. It is isomorphic to the induced map:

  $$F(A) \quad \to \quad F(\Gamma) \tag{3.8.27}$$

  which is a fibration because F preserves fibrations.

Finally pointwise limits commute strictly in:

$$\mathcal{C} \multimap \mathcal{D} \tag{3.8.28}$$

because limits commute strictly in $\mathcal{D}$. □

Remark 3.8.2. As for strict lex categories, considering strict clans is necessary for pointwise limits of morphisms to be morphisms. We could presumably use a 2-category of clans, with a weaker notion of morphism.

Now we define the tensor product of two strict clans.

Definition 3.8.3. Given strict clans $\mathcal{C}$ and $\mathcal{D}$, we define $\mathcal{C} \otimes \mathcal{D}$ as the strict clan freely generated by a functor:

$$\_ \otimes \_ \quad : \quad \mathcal{C} \times \mathcal{D} \to \mathcal{C} \otimes \mathcal{D} \tag{3.8.29}$$

such that:

- For any two fibrations:

  $$A \quad \twoheadrightarrow \quad \Gamma \tag{3.8.30}$$

  $$B \quad \twoheadrightarrow \quad \Delta \tag{3.8.31}$$

  in $\mathcal{C}$ and $\mathcal{D}$ we have an induced fibration:

  $$A \otimes B \quad \twoheadrightarrow \quad (\Gamma \otimes B) \underset{\Gamma \otimes \Delta}{\times} (A \otimes \Delta) \tag{3.8.32}$$

- For any objects Γ and Δ the induced morphisms:

  $$\Gamma \otimes \top \quad \to \quad \top \tag{3.8.33}$$

  $$\top \otimes \Delta \quad \to \quad \top \tag{3.8.34}$$

  are identities.

- Given spans:

$$A \twoheadrightarrow \Gamma_1 \leftarrow \Gamma_2$$

$$B \twoheadrightarrow \Delta_1 \leftarrow \Delta_2$$

with objects Γ and Δ, the induced morphisms:

$$(A \underset{\Gamma_1}{\times} \Gamma_2) \otimes \Delta \quad \to \quad (A \otimes \Delta) \underset{\Gamma_1 \otimes \Delta}{\times} (\Gamma_2 \otimes \Delta) \tag{3.8.35}$$

$$\Gamma \otimes (B \underset{\Delta_1}{\times} \Delta_2) \quad \to \quad (\Gamma \otimes B) \underset{\Gamma \otimes \Delta_1}{\times} (\Gamma \otimes \Delta_2) \tag{3.8.36}$$

are identities.

Check. For these axioms to make sense, we need to show that for any fibration:

$$A \twoheadrightarrow \Gamma \tag{3.8.37}$$

in $\mathcal{C}$ and $\Delta : \mathcal{D}$ we have an induced fibration:

$$A \otimes \Delta \quad \twoheadrightarrow \quad \Gamma \otimes \Delta \tag{3.8.38}$$

and the same with $\mathcal{C}$ and $\mathcal{D}$ reversed. This holds because:

- By the first assumption in $\mathcal{C} \otimes \mathcal{D}$ applied to:

$$A \twoheadrightarrow \Gamma \tag{3.8.39}$$

$$\Delta \twoheadrightarrow \top \tag{3.8.40}$$

we have an induced fibration:

$$A \otimes \Delta \quad \twoheadrightarrow \quad (\Gamma \otimes \Delta) \underset{\Gamma \otimes \top}{\times} (A \otimes \top) \tag{3.8.41}$$

- By the second assumption, this fibration is isomorphic to:

$$A \otimes \Delta \quad \twoheadrightarrow \quad \Gamma \otimes \Delta \tag{3.8.42}$$

The reverse is similar. □

Remark 3.8.4. In homotopy theory, given a tensor $\_ \otimes \_$ it is often assumed that given two cofibrations:

$$\Gamma \quad \rightarrowtail \quad \Gamma' \tag{3.8.43}$$

$$\Delta \quad \rightarrowtail \quad \Delta' \tag{3.8.44}$$

we have an induced cofibration:

$$(\Gamma \otimes \Delta') \coprod_{\Gamma \otimes \Delta} (\Gamma' \otimes \Delta) \quad \rightarrowtail \quad \Gamma' \otimes \Delta' \tag{3.8.45}$$

We end up with the dual condition because we used the opposite from the standard categories of cubes to begin with.

Finally we define the unit.

Definition 3.8.5. We define 1 as the free strict clan generated by an object.

Remark 3.8.6. The strict clan 1 is equivalent to the opposite of the category of finite sets, with monomorphisms as fibrations.

Now we are ready to prove that this defines a symmetric monoidal closed structure on the category of strict clans. We will proceed as for strict lex categories in Theorem 3.4.16, with a few more properties to check in order to take fibrations into account.

We denote by L the left adjoint to the forgetful functor U from strict clans to categories. First we give the variant of Lemma 3.4.14 for strict clans.

Lemma 3.8.7. *For any category $\mathcal{C}$ and strict clan $\mathcal{D}$ we have a natural isomorphism:*

$$U(L(\mathcal{C}) \multimap \mathcal{D}) \quad \cong \quad U(\mathcal{D})^{\mathcal{C}} \tag{3.8.46}$$

Moreover:

- *Fibrations in:*
$$L(\mathcal{C}) \multimap \mathcal{D} \tag{3.8.47}$$
correspond to pointwise fibrations in $U(\mathcal{D})^{\mathcal{C}}$.
- *Limits in:*
$$L(\mathcal{C}) \multimap \mathcal{D} \tag{3.8.48}$$
correspond to pointwise limits in $U(\mathcal{D})^{\mathcal{C}}$.

Proof. The natural isomorphism is very similar to the one for strict lex categories in Lemma 3.4.14, using pointwise fibrations in $\mathcal{D}^{\rightarrow}$.

- A new result we need to prove is that for any morphisms of strict clans:
$$F, G \quad : \quad L(\mathcal{C}) \multimap \mathcal{D} \tag{3.8.49}$$
with:
$$H \quad : \quad \mathrm{Hom}_{\mathrm{Cat}}(L(\mathcal{C}), \mathcal{D}^{\rightarrow}) \tag{3.8.50}$$
$$S \circ H \quad = \quad F \tag{3.8.51}$$
$$T \circ H \quad = \quad G \tag{3.8.52}$$
we have that H sends fibrations to pointwise fibrations in $\mathcal{D}^{\rightarrow}$. This means that for any fibration:
$$p \quad : \quad A \twoheadrightarrow \Gamma \tag{3.8.53}$$
in $L(\mathcal{C})$ we have that the vertical maps in:
$$\begin{tikzcd} F(A) \arrow[r, "H(A)"] \arrow[d, "F(p)"'] & G(A) \arrow[d, "G(p)"] \\ F(\Gamma) \arrow[r, "H(\Gamma)"'] & G(\Gamma) \end{tikzcd}$$
are fibrations. This holds because F and G preserve fibrations.

Now we want to check that fibrations in:

$$L(\mathcal{C}) \multimap \mathcal{D} \tag{3.8.54}$$

correspond to pointwise fibrations in $U(\mathcal{D})^{\mathcal{C}}$. Assume given:

$$F, G \quad : \quad L(\mathcal{C}) \multimap \mathcal{D} \tag{3.8.55}$$
$$\alpha \quad : \quad \mathrm{Hom}_{L(\mathcal{C}) \multimap \mathcal{D}}(F, G) \tag{3.8.56}$$

with:

$$\overline{F}, \overline{G} \quad : \quad U(\mathcal{D})^{\mathcal{C}} \tag{3.8.57}$$

$$\overline{\alpha} \quad : \quad \mathrm{Hom}_{U(\mathcal{D})^{\mathcal{C}}}(\overline{F}, \overline{G}) \tag{3.8.58}$$

the corresponding elements in $U(\mathcal{D})^{\mathcal{C}}$. The following are equivalent:

- The morphism α is a fibration.
- For all fibration:

  $$A \twoheadrightarrow \Gamma \tag{3.8.59}$$

  in $L(\mathcal{C})$, we have an induced fibration:

  $$F(A) \quad \twoheadrightarrow \quad G(A) \underset{G(\Gamma)}{\times} F(\Gamma) \tag{3.8.60}$$

- For all Γ and Δ in $L(\mathcal{C})$, we have an induced fibration:

  $$F(\Gamma) \times F(\Delta) \quad \twoheadrightarrow \quad F(\Gamma) \times G(\Delta) \tag{3.8.61}$$

  (because any fibration in $L(\mathcal{C})$ is isomorphic to a projection:

  $$\pi_1 \quad : \quad \Gamma \times \Delta \to \Gamma \tag{3.8.62}$$

  by Lemma 3.7.5, and we have that:

  $$\begin{tikzcd} F(\Gamma \times \Delta) \arrow[r] \arrow[d, "\cong"'] & G(\Gamma \times \Delta) \underset{G(\Gamma)}{\times} F(\Gamma) \arrow[d, "\cong"] \\ F(\Gamma) \times F(\Delta) \arrow[r] & F(\Gamma) \times G(\Delta) \end{tikzcd}$$

  so the top arrow is a fibration if and only if the bottom one is).
- For all Γ in $L(\mathcal{C})$ we have an induced fibration:

  $$F(\Gamma) \quad \twoheadrightarrow \quad G(\Gamma) \tag{3.8.63}$$

  (because given a fibration:

  $$F(\Gamma) \quad \twoheadrightarrow \quad G(\Gamma) \tag{3.8.64}$$

  for any $\Delta : \mathcal{D}$ we have an induced fibration:

  $$\Delta \times F(\Gamma) \quad \twoheadrightarrow \quad \Delta \times G(\Gamma) \tag{3.8.65}$$

  and we can conclude).
- For all Γ in $\mathcal{C}$ we have an induced fibration:

  $$\overline{F}(\Gamma) \quad \twoheadrightarrow \quad \overline{G}(\Gamma) \tag{3.8.66}$$

  (because F and G are extensions of $\overline{F}$ and $\overline{G}$ commuting with limits, so the condition implies that:

  $$F(\Delta) \quad \twoheadrightarrow \quad G(\Delta) \tag{3.8.67}$$

  for all:

  $$\Delta \quad = \quad \Gamma_1 \times \cdots \times \Gamma_n \tag{3.8.68}$$

  with $\Gamma_1, \cdots, \Gamma_n : \mathcal{C}$ and any object in $L(\mathcal{C})$ is of this form).
- The morphism $\overline{\alpha}$ is a pointwise fibration.

The fact that limits are computed pointwise is straightforward. □

THEOREM 3.8.8. *The arrow, tensor and unit from Definitions 3.8.1, 3.8.3 and 3.8.5 give a symmetric monoidal closed structure on the category of strict clans.*

PROOF. The proof is an extension of the proof of Theorem 3.4.16.

- First we prove that we have a natural isomorphism:
$$\mathrm{Hom}_{\mathrm{Clan}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{E}) \quad \cong \quad \mathrm{Hom}_{\mathrm{Clan}}(\mathcal{C}, \mathcal{D} \multimap \mathcal{E}) \tag{3.8.69}$$
We know from the lex case that giving a limit preserving functor:
$$F \quad : \quad \mathrm{Hom}_{\mathrm{Cat}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{E}) \tag{3.8.70}$$
is equivalent to giving a limit preserving:
$$\overline{F} \quad : \quad \mathrm{Hom}_{\mathrm{Cat}}(\mathcal{C}, \mathcal{D} \multimap \mathcal{E}) \tag{3.8.71}$$
so it is enough to check that F preserves fibrations if and only if so does $\overline{F}$.
  - By the definition of the tensor, we have that F preserves fibrations if for any fibrations:
$$A \twoheadrightarrow \Gamma \tag{3.8.72}$$
$$B \twoheadrightarrow \Delta \tag{3.8.73}$$
    we have an induced fibration:
$$F(A \otimes B) \quad \twoheadrightarrow \quad F((\Gamma \otimes B) \underset{\Gamma \otimes \Delta}{\times} (A \otimes \Delta)) \tag{3.8.74}$$
    which is isomorphic to:
$$F(A \otimes B) \quad \twoheadrightarrow \quad F(\Gamma \otimes B) \underset{F(\Gamma \otimes \Delta)}{\times} F(A \otimes \Delta) \tag{3.8.75}$$
  - On the other hand $\overline{F}$ preserves fibrations if for any fibration:
$$A \twoheadrightarrow \Gamma \tag{3.8.76}$$
    we have a fibration:
$$\overline{F}(A) \quad \twoheadrightarrow \quad \overline{F}(\Gamma) \tag{3.8.77}$$
    in $\mathcal{D} \multimap \mathcal{E}$, which means precisely that for any fibration:
$$B \twoheadrightarrow \Delta \tag{3.8.78}$$
    we have a fibration:
$$\overline{F}(A, B) \quad \twoheadrightarrow \quad \overline{F}(\Gamma, B) \underset{\overline{F}(\Gamma, \Delta)}{\times} \overline{F}(A, \Delta) \tag{3.8.79}$$

  So we see that both conditions are equivalent.
- Next we check symmetry. To extend the result for strict lex categories, we need to check that:
$$\mathrm{Sym} \quad : \quad \mathcal{C} \otimes \mathcal{D} \to \mathcal{D} \otimes \mathcal{C} \tag{3.8.80}$$
preserves fibrations. By the definition of the tensor, this holds if for any fibrations:
$$A \twoheadrightarrow \Gamma \tag{3.8.81}$$
$$B \twoheadrightarrow \Delta \tag{3.8.82}$$
we have an induced fibration:
$$\mathrm{Sym}(A \otimes B) \quad \twoheadrightarrow \quad \mathrm{Sym}((\Gamma \otimes B) \underset{\Gamma \otimes \Delta}{\times} (A \otimes \Delta)) \tag{3.8.83}$$

but this map is isomorphic to:

$$B \otimes A \quad \twoheadrightarrow \quad (\Delta \otimes A) \underset{\Delta \otimes \Gamma}{\times} (B \otimes \Gamma) \tag{3.8.84}$$

which is a fibration in $\mathcal{D} \otimes \mathcal{C}$.

- Then we check associativity. To extend the result for strict lex categories, we need to check that:

$$\mathrm{Assoc} \quad : \quad (\mathcal{C} \otimes \mathcal{D}) \otimes \mathcal{E} \to \mathcal{C} \otimes (\mathcal{D} \otimes \mathcal{E}) \tag{3.8.85}$$

preserves fibrations. By the definition of the tensor, this holds if for any fibrations:

$$\Gamma \twoheadrightarrow \Gamma' \tag{3.8.86}$$
$$\Delta \twoheadrightarrow \Delta' \tag{3.8.87}$$
$$\Theta \twoheadrightarrow \Theta' \tag{3.8.88}$$

we have an induced fibration:

$$\mathrm{Assoc}((\Gamma \otimes \Delta) \otimes \Theta) \quad \twoheadrightarrow \quad \mathrm{Assoc}\big(((\Gamma \otimes \Delta)' \otimes \Theta)'\big) \tag{3.8.89}$$

where we used the informal notation:

$$(X \otimes Y)' \quad = \quad X' \otimes Y \underset{X' \otimes Y'}{\times} X \otimes Y' \tag{3.8.90}$$

but the induced maps in:

$$(\Gamma \otimes \Delta) \otimes \Theta \quad \twoheadrightarrow \quad ((\Gamma \otimes \Delta)' \otimes \Theta)' \tag{3.8.91}$$

and:

$$\Gamma \otimes (\Delta \otimes \Theta) \quad \twoheadrightarrow \quad \big(\Gamma \otimes (\Delta \otimes \Theta)'\big)' \tag{3.8.92}$$

are isomorphic as their right-hand sides are the limits of the diagrams:

$$\begin{tikzcd}
(\Gamma' \otimes \Delta) \otimes \Theta \arrow[dr] \arrow[drr] & (\Gamma \otimes \Delta') \otimes \Theta \arrow[dl] \arrow[dr] & (\Gamma \otimes \Delta) \otimes \Theta' \arrow[dll] \arrow[dl] \\
(\Gamma \otimes \Delta') \otimes \Theta' & (\Gamma' \otimes \Delta) \otimes \Theta' & (\Gamma' \otimes \Delta') \otimes \Theta
\end{tikzcd}$$

and:

$$\begin{tikzcd}
\Gamma' \otimes (\Delta \otimes \Theta) \arrow[dr] \arrow[drr] & \Gamma \otimes (\Delta' \otimes \Theta) \arrow[dl] \arrow[dr] & \Gamma \otimes (\Delta \otimes \Theta') \arrow[dll] \arrow[dl] \\
\Gamma \otimes (\Delta' \otimes \Theta') & \Gamma' \otimes (\Delta \otimes \Theta') & \Gamma' \otimes (\Delta' \otimes \Theta)
\end{tikzcd}$$

which are isomorphic using the associativity of $\_ \otimes \_$.

- Finally we check the unit isomorphisms using the fact that:

$$(1 \multimap \mathcal{C}) \quad \cong \quad \mathcal{C} \tag{3.8.93}$$

by Lemma 3.8.7 applied to $1 = L(\top)$ with $\top$ the terminal category.

$\square$

3.9. Reedy fibrant cubical objects

Now we can apply the machinery from Section 3.2 to define notions of parametricity for strict clans as monoids in the category of strict clans. If we unfold this we get:

Proposition 3.9.1. *A notion of parametricity for strict clans is a strict clan $\mathcal{M}$ with a strict monoidal product:*

$$\_ \otimes \_ \quad : \quad \mathcal{M} \times \mathcal{M} \to \mathcal{M} \tag{3.9.1}$$

such that:

- *For any fibrations:*
$$A \twoheadrightarrow A' \tag{3.9.2}$$
$$B \twoheadrightarrow B' \tag{3.9.3}$$
we have an induced fibration:
$$A \otimes B \twoheadrightarrow (A' \otimes B) \underset{A' \otimes B'}{\times} (A \otimes B') \tag{3.9.4}$$
- *For any object $\Gamma : \mathcal{C}$ the induced morphisms:*
$$\Gamma \otimes \top \to \top \tag{3.9.5}$$
$$\top \otimes \Gamma \to \top \tag{3.9.6}$$
are identities.
- *Given a span:*
$$A \twoheadrightarrow \Gamma_1 \longleftarrow \Gamma_2$$
in $\mathcal{C}$ with $\Delta : \mathcal{C}$, the induced morphisms:
$$(A \underset{\Gamma_1}{\times} \Gamma_2) \otimes \Delta \to (A \otimes \Delta) \underset{\Gamma_1 \otimes \Delta}{\times} (\Gamma_2 \otimes \Delta) \tag{3.9.7}$$
$$\Delta \otimes (A \underset{\Gamma_1}{\times} \Gamma_2) \to (\Delta \otimes A) \underset{\Delta \otimes \Gamma_1}{\times} (\Delta \otimes \Gamma_2) \tag{3.9.8}$$
are identities.

Now we give some examples. First we extend notions of parametricity for categories to strict clans.

Proposition 3.9.2. *The functor L freely adding a strict clan structure to a category is strongly monoidal, meaning that we have natural isomorphisms:*

$$L(\mathcal{C} \times \mathcal{D}) \cong L(\mathcal{C}) \otimes L(\mathcal{D}) \tag{3.9.9}$$
$$L(\top) \cong 1 \tag{3.9.10}$$

obeying some coherence conditions.

Proof. This is exactly the same as Lemma 3.5.3 for strict lex categories, using Lemma 3.8.7 instead of Lemma 3.4.14. □

Corollary 3.9.3. *If $\mathcal{C}$ is a notion of parametricity for categories, then $L(\mathcal{C})$ is a notion of parametricity for strict clans.*

Proof. Strongly monoidal functors preserve monoids. □

We give an explicit description for the cofreely parametric strict clans build from a notion of parametricity for categories:

Proposition 3.9.4. *The strict clan:*

$$L(\mathcal{C}) \multimap \mathcal{D} \tag{3.9.11}$$

is isomorphic to the strict clan where:

- *Objects are functors from $\mathcal{C}$ to $\mathcal{D}$.*
- *Morphisms are natural transformations.*
- *Limits and fibrations are defined pointwise.*

Proof. This is a reformulation of Lemma 3.8.7. $\square$

Now we introduce some notions of parametricity exclusive to clans. They depend on a suitable monoidal category, called a category with a cubical interval.

Definition 3.9.5. A category with a cubical interval is a monoidal category $\square$ with:

$$\mathbb{I} \;:\; \square \tag{3.9.12}$$

$$d^0, d^1 \;:\; \mathbb{I} \to 1 \tag{3.9.13}$$

such that any object in $\square$ is of the form $\mathbb{I}^n$ for some $n \geq 0$.

Having a cubical interval is a very minimal requirement for a category to be called a category of cubes. All (the opposite of) the examples from [**BM17**] are categories with cubical intervals.

Definition 3.9.6. Assume given $\square$ a category with a cubical interval, then we define a notion of parametricity for strict clans $\square_c$ as the monoidal strict clan freely generated by:

- A monoidal functor from $\square$ to $\square_c$
- The fact that we have an induced fibration:

  $$(d^0, d^1) \;:\; \mathbb{I} \twoheadrightarrow 1 \times 1 \tag{3.9.14}$$

  in $\square_c$

Definition 3.9.7. We call $\square_c$ the cubical notion of parametricity associated to $\square$.

Example 3.9.8. The main (and simplest) example is the monoidal strict clan freely generated by an object $\mathbb{I}$ and two maps:

$$d^0, d^1 \;:\; \mathbb{I} \to 1 \tag{3.9.15}$$

inducing a fibration:

$$(d^0, d^1) \;:\; \mathbb{I} \twoheadrightarrow 1 \times 1 \tag{3.9.16}$$

It is called the strict clan of semi-cubes.

The previous example can be adapted to any variant of cubes.

Remark 3.9.9. The strict clan of semi-cubes $\square_c$ gives the standard notion of parametricity. Indeed we have a fibration:

$$(d^0, d^1) \;:\; \mathbb{I} \twoheadrightarrow 1 \times 1 \tag{3.9.17}$$

so that in any $\square_c$-parametric strict clan $\mathcal{C}$ we have that for any fibration:

$$A \twoheadrightarrow \Gamma \tag{3.9.18}$$

we have an induced fibration:

$$A_* \quad \twoheadrightarrow \quad (A \times A) \underset{\Gamma\times\Gamma}{\times} \Gamma_* \tag{3.9.19}$$

This is analogous to the standard parametricity from Definition 1.3.1, up the the strictness assumptions.

We give an auxiliary definition.

Definition 3.9.10. Given two fibrations in a clan:

$$\sigma \quad : \quad \Gamma \twoheadrightarrow \Gamma' \tag{3.9.20}$$

$$\delta \quad : \quad \Delta \twoheadrightarrow \Delta' \tag{3.9.21}$$

we define $\sigma \odot \delta$ using the pullback square:

$$\begin{tikzcd}
& \Gamma\otimes\Delta \arrow[d, "\sigma\odot\delta"] \arrow[ddl, bend right, "\sigma\otimes\Delta"'] \arrow[ddr, bend left, "\Gamma\otimes\delta"] & \\
& (\Gamma'\otimes\Delta) \underset{\Gamma'\otimes\Delta'}{\times} (\Gamma\otimes\Delta') \arrow[dl] \arrow[dr] & \\
\Gamma'\otimes\Delta \arrow[dr, "\Gamma'\otimes\delta"'] & & \Gamma\otimes\Delta' \arrow[dl, "f\otimes\Delta'"] \\
& \Gamma'\otimes\Delta' &
\end{tikzcd}$$

We define the dual to borders from Definition 3.6.2.

Definition 3.9.11. Given a strict clan $\mathcal{C}$ with a fibration:

$$i \quad : \quad A \twoheadrightarrow \Gamma \tag{3.9.22}$$

we define:

$$\delta A^n \quad : \quad \mathrm{Ob}_{\mathcal{C}} \tag{3.9.23}$$

$$\delta i^n \quad : \quad A^n \twoheadrightarrow \delta A^n \tag{3.9.24}$$

inductively on n by:

$$\delta A^0 \quad = \quad \top \tag{3.9.25}$$

$$\delta i^0 \quad = \quad \epsilon_1 \tag{3.9.26}$$

with:

$$\delta A^{n+1} \quad = \quad (\delta A^n) \otimes A \underset{(\delta A^n)\otimes\Gamma}{\times} A^n \otimes \Gamma \tag{3.9.27}$$

$$\delta i^{n+1} \quad = \quad \delta i^n \odot i \tag{3.9.28}$$

Now we can give the key lemma for cubical notions of parametricity.

Lemma 3.9.12. *Assume given $\square_c$ the cubical notion of parametricity associated to a category $\square$ with a cubical interval. The strict clan underlying $\square_c$ is isomorphic to the strict clan $\square_\delta$ freely generated by:*

- *A functor from the category $\square$ to $\square_\delta$.*
- *The fact that δi^n is a fibration in $\square_\delta$ for all $n \geq 0$, where:*

$$i \quad = \quad (d^0, d^1) \tag{3.9.29}$$

Proof. We will prove that both strict clans are isomorphic to an intermediate strict clan $\square'$ freely generated by:

- A morphism of strict clans from $L_+(\square)$ to $\square'$. Here $L_+(\square)$ is the monoidal strict clan freely generated by the monoidal category $\square$.
- The fact that δi^n is a fibration in $\square'$ for all $n \geq 0$, where:

$$i \quad = \quad (d^0, d^1) \tag{3.9.30}$$

First we prove that $\square'$ and $\square_\delta$ are isomorphic:

- The following square commutes up to natural isomorphism:

$$\begin{tikzcd} \mathrm{Cat} \arrow[d, "L"'] & \mathrm{MonCat} \arrow[l, "U_+"'] \arrow[d, "L_+"] \\ \mathrm{Clan} & \mathrm{MonClan} \arrow[l, "U"] \end{tikzcd}$$

because L is strongly monoidal by Lemma 3.9.2. So we have natural isomorphisms:

$$\begin{aligned} \mathrm{Hom}_{\mathrm{Clan}}(UL_+(\square), \mathcal{D}) &\cong \mathrm{Hom}_{\mathrm{Clan}}(LU_+(\square), \mathcal{D}) & (3.9.31) \\ &\cong \mathrm{Hom}_{\mathrm{Cat}}(U_+(\square), \mathcal{D}) & (3.9.32) \end{aligned}$$

- This means that a morphism of strict clans from $L_+(\square)$ to $\mathcal{D}$ is equivalent to a functor from $\square$ to $\mathcal{D}$. So $\square'$ is isomorphic to $\square_\delta$.

To prove that $\square_c$ and $\square'$ are isomorphic and conclude, it is enough to prove that in $L_+(\square)$ the following conditions are equivalent:

(1) The map δi^n is a fibration for any $n \geq 0$.
(2) The map i is a fibration and fibrations are stable under $\_ \odot \_$.

We have that (2) implies (1) because δi^0 is always a fibration and:

$$\delta i^{n+1} \quad = \quad \delta i^n \odot i \tag{3.9.33}$$

To prove that (1) implies (2) we assume that δi^n is a fibration for all $n \geq 0$. The map i is a fibration as it is isomorphic to δi^1. So we just need to check that fibrations are closed by $\_ \odot \_$.

We prove inductively on two fibrations f, g in $L_+(\square)$ that the map $f \odot g$ is a fibration. At first we fix:

$$f \quad : \quad X \twoheadrightarrow Y \tag{3.9.34}$$

and do the induction on g.

- Given fibrations:

$$\begin{aligned} q \quad &: \quad B \twoheadrightarrow A & (3.9.35) \\ p \quad &: \quad A \twoheadrightarrow \Gamma & (3.9.36) \end{aligned}$$

we have a commutative square:

$$\begin{tikzcd} X \otimes B \arrow[r, "f \odot (p \circ q)"] \arrow[d, "f \odot q"'] & (Y \otimes B) \underset{Y \otimes \Gamma}{\times} (X \otimes \Gamma) \arrow[d, "\cong"] \\ (Y \otimes B) \underset{Y \otimes A}{\times} (X \otimes A) \arrow[r, "(Y \otimes B) \underset{Y \otimes A}{\times} (f \odot p)"'] & (Y \otimes B) \underset{Y \otimes A}{\times} (Y \otimes A) \underset{Y \otimes \Gamma}{\times} (X \otimes \Gamma) \end{tikzcd}$$

so that if $f \odot p$ and $f \odot q$ are fibrations, then so is $f \odot (p \circ q)$.

- Given:

$$
\begin{aligned}
p &: A \twoheadrightarrow \Gamma && (3.9.37)\\
\sigma &: \Delta \to \Gamma && (3.9.38)
\end{aligned}
$$

we have a pullback square:

$$
\begin{tikzcd}
A \times_{\Gamma} \Delta \arrow[r, two heads, "\pi_2"] \arrow[d] & \Delta \arrow[d, "\sigma"] \\
A \arrow[r, two heads, "p"'] & \Gamma
\end{tikzcd}
$$

and a commutative diagram:

$$
\begin{tikzcd}
X \otimes (A \times_{\Gamma} \Delta) \arrow[rr, "f \odot \pi_2"] \arrow[d, "\cong"'] & & (Y \otimes (A \times_{\Gamma} \Delta)) \times_{Y \otimes \Delta} (X \otimes \Delta) \arrow[d, "\cong"] \\
(X \otimes A) \times_{X \otimes \Gamma} (X \otimes \Delta) \arrow[d, "(f \odot p) \times_{X \otimes \Gamma} (X \otimes \Delta)"'] & & (Y \otimes A) \times_{Y \otimes \Gamma} (Y \otimes \Delta) \times_{Y \otimes \Delta} X \otimes \Delta \arrow[d, "\cong"] \\
(Y \otimes A) \times_{Y \otimes \Gamma} (X \otimes \Gamma) \times_{X \otimes \Gamma} (X \otimes \Gamma') \arrow[rr, "\cong"'] & & (Y \otimes A) \times_{Y \otimes \Gamma} (X \otimes \Delta)
\end{tikzcd}
$$

so that if $f \odot p$ is a fibration, then so is $f \odot \pi_2$.

- For Γ an object, we have a fibration:

$$
\epsilon_\Gamma \quad : \quad \Gamma \twoheadrightarrow \top \qquad (3.9.39)
$$

with a commutative square:

$$
\begin{tikzcd}
X \otimes \Gamma \arrow[r, "f \odot \epsilon_\Gamma"] \arrow[d, "f \otimes \Gamma"'] & (Y \otimes \Gamma) \times_{Y \otimes \top} (X \otimes \top) \arrow[d, "\cong"] \\
Y \otimes \Gamma \arrow[r, "\cong"'] & (Y \otimes \Gamma) \times_{\top} \top
\end{tikzcd}
$$

So it is enough to prove that $f \otimes \Gamma$ is a fibration for all Γ. We proceed inductively on Γ, using the following isomorphisms:

$$
\begin{aligned}
f \otimes 1 &\cong f && (3.9.40)\\
f \otimes (\Gamma \otimes \Delta) &\cong (f \otimes \Gamma) \otimes \Delta && (3.9.41)\\
f \otimes \top &\cong \mathrm{id}_\top && (3.9.42)\\
f \otimes (A \times_{\Gamma} \Delta) &\cong (f \otimes A) \times_{f \otimes \Gamma} (f \otimes \Delta) && (3.9.43)
\end{aligned}
$$

Objects are generated by $\mathbb{I}$ in a category with cubical interval, so we just need to prove that $f \otimes \mathbb{I}$ is a fibration to conclude. We have a commutative triangle:

$$\begin{tikzcd}
X \otimes \mathbb{I} \arrow[rr, "f \otimes \mathbb{I}"] \arrow[dr, "f \odot i"'] & & Y \otimes \mathbb{I} \\
& (Y \otimes \mathbb{I}) \underset{Y \times Y}{\times} (X \times X) \arrow[ur, "\pi_1"'] &
\end{tikzcd}$$

where π_1 is a pullback of the fibration:

$$f \times f \quad : \quad X \times X \twoheadrightarrow Y \times Y \tag{3.9.44}$$

so that if $f \odot i$ is a fibration then so is $f \otimes \mathbb{I}$.

So the only case remaining is when $g = i$.

Iterating the previous isomorphisms (and the analogous ones for the left variable f), we can assume that $f \odot g$ is build from i and $\_ \odot \_$ only. Since $\_ \odot \_$ is associative this means that $f \odot g$ is isomorphic to δi^n from some $n > 0$, so it is a fibration. □

Now we want to analyse cofreely parametric models for a cubical notion of parametricity. We need an auxiliary definition:

Definition 3.9.13. Let $\mathcal{C}$ be a strict clan, and let $\square$ be a category with a cubical interval.

- A functor $F : \mathcal{C}^\square$ is called cubically fibrant if for all $n \geq 0$ we have an induced fibration:
$$F(\mathbb{I}^n) \twoheadrightarrow \widetilde{F}(\delta\mathbb{I}^n) \tag{3.9.45}$$
where $\widetilde{F}$ extends F by commuting with limits.
- A morphism:
$$\alpha \quad : \quad F \to G \tag{3.9.46}$$
in $\mathcal{C}^\square$ with F to G cubically fibrant is called a cubical fibration if for all $n \geq 0$ we have a Reedy fibrant square:
$$\begin{tikzcd}
F(\mathbb{I}^n) \arrow[r, "\alpha(\mathbb{I}^n)"] \arrow[d] & G(\mathbb{I}^n) \arrow[d] \\
\widetilde{F}(\delta\mathbb{I}^n) \arrow[r, "\widetilde{\alpha}(\delta\mathbb{I}^n)"'] & \widetilde{G}(\delta\mathbb{I}^n)
\end{tikzcd}$$
where $\widetilde{\alpha}$ (resp. $\widetilde{F}, \widetilde{G}$) extends α (resp. F, G) by commuting with limits.

Remark 3.9.14. It should be noted that:

$$\widetilde{F}(\delta\mathbb{I}^n) \tag{3.9.47}$$

can be defined inductively by commuting with limits only if the maps:

$$F(\mathbb{I}^k) \to \widetilde{F}(\delta\mathbb{I}^k) \tag{3.9.48}$$

are fibrations for $k < n$.

Remark 3.9.15. Assume given $\square$ a Reedy category of cubes where faces are the only dimension-decreasing maps, for example semi-cubes, or cubes with reflexivities. Then we have that:

- Cubically fibrant objects are precisely Reedy fibrant objects.
- Cubical fibrations are precisely Reedy fibrations.

Indeed the matching object of F at $\mathbb{I}^n$ is by definition $\widetilde{F}$ applied to the border of $\mathbb{I}^n$. But when faces are the only dimension-decreasing maps, the border of $\mathbb{I}^n$ is isomorphic to $\delta\mathbb{I}^n$, so we can conclude.

Proposition 3.9.16. *Let $\mathcal{C}$ be a strict clan, and $\square_c$ be the cubical notion of parametricity associated to a category $\square$ with a cubical interval. Then the strict clan:*

$$\square_c \multimap \mathcal{C} \tag{3.9.49}$$

is isomorphic to the strict clan of cubically fibrant objects in $\mathcal{C}^{\square}$ equipped with cubical fibrations.

Proof. By Lemma 3.9.12 giving an object in:

$$\square_c \multimap \mathcal{C} \tag{3.9.50}$$

is the same as giving a cubically fibrant object in $\mathcal{C}^{\square}$.

Next we prove that fibrations in:

$$\square_c \multimap \mathcal{C} \tag{3.9.51}$$

are precisely cubical fibrations. We define Δ as the strict clan freely generated by two objects and a fibration between them. For $\mathcal{D}$ a strict clan we have that:

$$\Delta \multimap \mathcal{D} \tag{3.9.52}$$

is such that:

- Its objects are fibrations in $\mathcal{D}$.
- Its morphisms are commutative squares.
- Its fibrations are Reedy fibrant squares.

Then we have an isomorphism:

$$\mathrm{Hom}_{\mathrm{Clan}}(\Delta, \square_c \multimap \mathcal{C}) \quad \cong \quad \mathrm{Hom}_{\mathrm{Clan}}(\square_c, \Delta \multimap \mathcal{C}) \tag{3.9.53}$$

so that, by Lemma 3.9.12, a fibration in:

$$\square_c \multimap \mathcal{C} \tag{3.9.54}$$

is the same thing as:

- A functor from $\square$ to:

  $$\Delta \multimap \mathcal{C} \tag{3.9.55}$$

  This is equivalent to a pointwise fibration:

  $$\alpha \quad : \quad F \to G \tag{3.9.56}$$

  in $\mathcal{C}^{\square}$.
- Such that for any $n \geq 0$ we have a Reedy fibrant square:

  $$\begin{tikzcd} F(\mathbb{I}^n) \arrow[r, "\alpha(\mathbb{I}^n)"] \arrow[d] & G(\mathbb{I}^n) \arrow[d] \\ \widetilde{F}(\delta\mathbb{I}^n) \arrow[r, "\widetilde{\alpha}(\delta\mathbb{I}^n)"'] & \widetilde{G}(\delta\mathbb{I}^n) \end{tikzcd}$$

  where $\widetilde{\alpha}$ (resp. $\widetilde{F}, \widetilde{G}$) extends α (resp. F, G) by commuting with limits.

But this is precisely a cubical fibration. ◻

Remark 3.9.17. We asserted without proof that cubically fibrant F in $\mathcal{C}^{\square}$ with cubical fibrations form a strict clan. This is in fact a consequence of Proposition 3.9.16.

Remark 3.9.18. By Remark 3.9.15 and Proposition 3.9.16, we can conclude that clans of Reedy fibrant semi-cubical (or cubical with reflexivities) objects are cofreely parametric.

We conjecture that this result can be extended to any Reedy category with a suitable monoidal structure, using the monoidal strict clan generated by $\square$ with dimension-decreasing maps as fibrations.

Conclusion

We defended the thesis that cubical models are in fact cofreely parametric. To do this we offered two frameworks for variants of parametricity. In both framework, functors forgetting parametricity have right (and left) adjoints, so that cofreely (and freely) parametric models exist. We compare these frameworks:

- **Extensions by sections.** (Chapter 2)
  - Properly defining extensions by section required the theory of signatures for quotient inductive-inductive types.
  - There are many examples of extensions by section, and a lot of them have intuitively very little in common with parametricity, for example groups extending monoids. It excludes desirable notions of parametricity, including truncated forms of parametricity.
  - Constructing examples of extension by section is relatively straightforward, but sometimes tedious (e.g. in Appendix A).
  - The formula giving cofreely parametric models for an extension by section is usually not convenient to work with.
- **Monoids and modules.** (Chapter 3)
  - This framework is a lot simpler to present than extensions by section, as it uses only elementary categorical notions.
  - Module structures and extensions by section are incomparable. Some desirable examples are excluded by modules, notably Kan fibrations in clans.
  - Giving a notion of parametricity (i.e. giving a monoidal structure on a model) is usually difficult, at least when the model was not expressly build as monoidal.
  - The formula giving cofreely parametric models as coinduced modules is convenient, allowing us to prove that many cubical models are in fact cofreely parametric.

We examine some potential further works:

- Using *strict* lex categories and *strict* clans significantly damped the applicability of Chapter 3. To solve this, it is natural to consider the 2-categories of lex categories and clans. Then the whole chapter could be reworked using 2-categorical notions.

  For example we would define notions of parametricity as monoids with associativity and unitality holding only up to isomorphisms, giving *non-strict* monoidal models. Using weak morphisms, we would get notions of parametricity where the isomorphisms:

$$(A \times B)_* \quad \cong \quad A_* \times B_* \tag{3.9.57}$$

and other similar rules are *not* equality. So these notions are technically not extensions by section, although they should be 2-categorical extensions by section in some sense.

- Chapter 3 required a symmetric monoidal closed category of models of type theory, so it is desirable to have more examples of such categories. In particular we should get a symmetric monoidal closed category of categories with families (either adding strictness assumptions, or using a 2-category).

  We conjecture that the lex category T classifying categories with families is monoidal lex, and that we can build tensors and arrows for category with families as a variants of Day convolution adapted to lex categories. We conjecture that this holds for any notion of model of type theory without arrow types or a universe.

Remark 3.9.19. Assume given a lex category T classifying some notion of model of type theory, meaning that:

$$\mathrm{Hom}_{\mathrm{Lex}}(T, \mathrm{Set}) \quad \cong \quad \{\text{models of type theory}\} \tag{3.9.58}$$

If T is lex monoidal, we have a natural transformations:

$$\begin{aligned} \mathrm{Hom}_{\mathrm{Lex}}(T, \mathrm{Set}) \quad &\to \quad \mathrm{Hom}_{\mathrm{Lex}}(T \otimes T, \mathrm{Set}) && (3.9.59)\\ &\cong \quad \mathrm{Hom}_{\mathrm{Lex}}(T, T \multimap \mathrm{Set}) && (3.9.60)\\ &\cong \quad \mathrm{Hom}_{\mathrm{Lex}}(T, \{\text{models of type theory}\}) && (3.9.61) \end{aligned}$$

This means that any model of type theory can be seen as a model *internal to models of type theory*, in the same way that any category can be seen as a *double category* (i.e. a category internal to categories). We believe that this is a crucial propriety of models for type theory.

Remark 3.9.20. Intriguingly, this also means that T is a notion of parametricity for lex categories! Then:

- A lex category $\mathcal{C}$ is T-parametric if any object Γ in $\mathcal{C}$ is the object of contexts of a model of type theory internal to $\mathcal{C}$, and any morphism in $\mathcal{C}$ extends to a morphism of models.
- The cofreely T-parametric lex category generated by a lex category $\mathcal{C}$ is the category of models of type theory internal to $\mathcal{C}$.

- We did not prove that Kan cubical structures can be generated as cofreely parametric. Neither framework are appropriate:
  - It is not possible to generate Kan cubical structures in the module framework. Indeed say we have $\mathcal{M}$ a notion of parametricity for clans. Then fibrations in a cofreely parametric model:

$$\mathcal{M} \multimap \mathcal{C} \tag{3.9.62}$$

    cannot be Kan fibrations, as they are defined by the condition that a bunch of morphisms in $\mathcal{C}$ are fibrations, and not by the existence of some liftings.
  - We can add coercions in an extension by sections, presumably generating Kan cubical structures. But since we do not have a convenient description for cofreely parametric models in this case, we do not know how to prove this.

A solution to this conundrum might be to use two successive extensions:

(1) An extension by a module structure, building cubical objects.
(2) An extension by the assumption that fibrations have liftings. It might be possible to prove that this is an extension by section and compute the right adjoint using the fact that Kan fibrations are stable under type constructors.

- A crucial limitation for both frameworks is likely incompatibility of internal parametricity with arrow types or a universe, as evoked in Remark 2.7.8. For categories we have the following:

  Lemma 3.9.21. *Assume given $\mathcal{C}$ and $\mathcal{D}$ two categories. If $\mathcal{D}$ has exponentials and* enough limits*, then $\mathcal{D}^{\mathcal{C}}$ has exponentials.*

  We hope that similar results hold for models of type theory:

  Conjecture 3.9.22. *Assume given $\mathcal{C}$ and $\mathcal{D}$ two models of type theory. If $\mathcal{D}$ has arrow types (resp. a universe) and* enough inductive types*, then:*

  $$\mathcal{C} \multimap \mathcal{D} \tag{3.9.63}$$

  has arrow types (resp. a universe).

  This approach would not provide computation rules for internal parametricity with arrow types and a universe.

- There is a large literature on forgetful functors having left or right adjoints (see for example [**AP03**]), but not on forgetful functors having both. While extensions by section provide many interesting examples of such functors, they do not give all of them.

  Indeed such functors will only add unary operations (or equations), but they do not need to be inductively defined (or proven) on old unary operations. As an example consider the extension of:

  $$X \quad : \quad \mathcal{U} \tag{3.9.64}$$
  $$s \quad : \quad X \to X \tag{3.9.65}$$

  by:

  $$\_{*} \quad : \quad X \to X \tag{3.9.66}$$

  This is not an extension by section as there is no equation defining $s(x)_*$. Nevertheless the associated forgetful functor:

  $$U \quad : \quad (X : \mathrm{Set},\ s, t : X \to X) \to (X : \mathrm{Set},\ s : X \to X) \tag{3.9.67}$$

  has a right adjoint.

  Remark 3.9.23. This can be generalized, indeed given any functor:

  $$F \quad : \quad I \to J \tag{3.9.68}$$

  the induced forgetful functor:

  $$F^{*} \quad : \quad \mathrm{Set}^{J} \to \mathrm{Set}^{I} \tag{3.9.69}$$

  has both left and right adjoints. Our example is the case where I (resp. J) is the category freely generated by an object and an endomorphism (resp. two endomorphisms) and F is an inclusion.

It would be interesting to give a syntactical necessary and sufficient condition for extensions to have both left and right adjoints. This is clearly linked to the problem of finding a syntax for higher coinductive types.

Remark 3.9.24. Maybe forgetful functors having both left and right adjoints should be called *unary*, or perhaps *linear*.

Indeed consider the forgetful functor sending some uni-sorted algebras to their underlying sets. This functor has both left and right adjoints if and only if the algebras can be defined using only unary operations and unary equations.

Bibliography

[ABKT19] Thorsten Altenkirch, Simon Boulier, Ambrus Kaposi, and Nicolas Tabareau. Setoid type theory—a syntactic translation. In *International Conference on Mathematics of Program Construction*, 2019.

[Adá74] Jiří Adámek. Free algebras and automata realizations in the language of categories. *Commentationes Mathematicae Universitatis Carolinae*, 1974.

[AGJ14] Robert Atkey, Neil Ghani, and Patricia Johann. A relationally parametric model of dependent type theory. In *Symposium on Principles of Programming Languages*, 2014.

[AHR99] Jirı Adámek, Michel Hébert, and Jirı Rosickỳ. On essentially algebraic theories and their generalizations. *Algebra Universalis*, 1999.

[AK16] Thorsten Altenkirch and Ambrus Kaposi. Type theory in type theory using quotient inductive types. *ACM SIGPLAN Notices*, 2016.

[AK17] Thorsten Altenkirch and Ambrus Kaposi. Towards a cubical type theory without an interval. *Leibniz International Proceedings in Informatics*, 2017.

[AP03] Jirí Adámek and Hans Porst. On varieties and covarieties in a category. *Mathematical Structures in Computer Science*, 2003.

[AR94] Jiří Adámek and Jiří Rosický. *Locally presentable and accessible categories*. Cambridge University Press, 1994.

[Awo18] Steve Awodey. Natural models of homotopy type theory. *Mathematical Structures in Computer Science*, 2018.

[BCM15] Jean-Philippe Bernardy, Thierry Coquand, and Guilhem Moulin. A presheaf model of parametric type theory. *Electronic Notes in Theoretical Computer Science*, 2015.

[BJP10] Jean-Philippe Bernardy, Patrik Jansson, and Ross Paterson. Parametricity and dependent types. In *International conference on Functional Programming*, 2010.

[BL11] Jean-Philippe Bernardy and Marc Lasson. Realizability and parametricity in pure type systems. In *International Conference on Foundations of Software Science and Computational Structures*, 2011.

[BM17] Ulrik Buchholtz and Edward Morehouse. Varieties of cubical sets. In *International Conference on Relational and Algebraic Methods in Computer Science*, 2017.

[Bra14] Martin Brandenburg. Tensor categorical foundations of algebraic geometry. *arXiv preprint arXiv:1410.1716*, 2014.

[Bro73] Kenneth Brown. Abstract homotopy theory and generalized sheaf cohomology. *Transactions of the American Mathematical Society*, 1973.

[Car86] John Cartmell. Generalised algebraic theories and contextual categories. *Annals of Pure and Applied Logic*, 1986.

[CCD21] Simon Castellan, Pierre Clairambault, and Peter Dybjer. Categories with families: Unityped, simply typed, and dependently typed. In *Joachim Lambek: The Interplay of Mathematics, Logic, and Linguistics*. Springer, 2021.

[CCHM15] Cyril Cohen, Thierry Coquand, Simon Huber, and Anders Mörtberg. Cubical type theory: a constructive interpretation of the univalence axiom. In *International Conference on Types for Proofs and Programs*, 2015.

[CD14] Pierre Clairambault and Peter Dybjer. The biequivalence of locally cartesian closed categories and martin-löf type theories. *Mathematical Structures in Computer Science*, 2014.

[CH20] Evan Cavallo and Robert Harper. Internal parametricity for cubical type theory. In *Annual Conference on Computer Science Logic*, 2020.

[Dyb95] Peter Dybjer. Internal type theory. In *International Workshop on Types for Proofs and Programs*, 1995.

[Fra11] Ignacio López Franco. Pseudo-commutativity of kz 2-monads. *Advances in Mathematics*, 2011.

[Fre66] Peter Freyd. Algebra valued functors in general and tensor products in particular. In *Colloquium Mathematicum*, 1966.

[GFO16] Neil Ghani, Fredrik Nordvall Forsberg, and Federico Orsanigo. Proof-relevant parametricity. In *A List of Successes That Can Change the World*. Springer, 2016.

[Hof94] Martin Hofmann. On the interpretation of type theory in locally cartesian closed categories. In *International Workshop on Computer Science Logic*, 1994.

[Hof97] Martin Hofmann. Syntax and semantics of dependent types. In *Extensional Constructs in Intensional Type Theory*. Springer, 1997.

[Hov07] Mark Hovey. *Model categories*. American Mathematical Society, 2007.

[HP02] Martin Hyland and John Power. Pseudo-commutative monads and pseudo-closed 2-categories. *Journal of Pure and Applied Algebra*, 2002.

[HPP06] Martin Hyland, Gordon Plotkin, and John Power. Combining effects: Sum and tensor. *Theoretical Computer Science*, 2006.

[Jac93] Bart Jacobs. Comprehension categories and the semantics of type dependency. *Theoretical Computer Science*, 1993.

[Joy17] André Joyal. Notes on clans and tribes. *arXiv preprint arXiv:1710.10238*, 2017.

[JS17] Patricia Johann and Kristina Sojakova. Cubical categories for higher-dimensional parametricity. *arXiv preprint arXiv:1701.06244*, 2017.

[Kel80] Max Kelly. A unified treatment of transfinite constructions for free algebras, free monoids, colimits, associated sheaves, and so on. *Bulletin of the Australian Mathematical Society*, 1980.

[KK20] András Kovács and Ambrus Kaposi. Signatures and induction principles for higher inductive-inductive types. *Logical Methods in Computer Science*, 2020.

[KKA19] Ambrus Kaposi, András Kovács, and Thorsten Altenkirch. Constructing quotient inductive-inductive types. *Proceedings of the ACM on Programming Languages*, 2019.

[KL12a] Chris Kapulkin and Peter LeFanu Lumsdaine. The simplicial model of univalent foundations (after voevodsky). *arXiv preprint arXiv:1211.2851*, 2012.

[KL12b] Chantal Keller and Marc Lasson. Parametricity in an impredicative sort. *Computer Science Logic*, 2012.

[KRRZ11] Carolyn Kennett, Emily Riehl, Michael Roy, and Michael Zaks. Levels in the toposes of simplicial sets and cubical sets. *Journal of Pure and Applied Algebra*, 2011.

[Law63] William Lawvere. *Functorial semantics of algebraic theories and Some algebraic problems in the context of functorial semantics of algebraic theories*. PhD thesis, Columbia University, 1963.

[Lur09] Jacob Lurie. *Higher topos theory*. Princeton University Press, 2009.

[LW15] Peter LeFanu Lumsdaine and Michael Warren. The local universes model: an overlooked coherence construction for dependent type theories. *Transactions on Computational Logic*, 2015.

[ML75] Per Martin-Löf. An intuitionistic theory of types: Predicative part. In *Studies in Logic and the Foundations of Mathematics*. Elsevier, 1975.

[MLS84] Per Martin-Löf and Giovanni Sambin. *Intuitionistic type theory*. Bibliopolis, 1984.

[Moe21] Hugo Moeneclaey. Parametricity and semi-cubical types. In *Symposium on Logic in Computer Science*, 2021.

[Par77] Bodo Pareigis. Non-additive ring and module theory. *Publicationes mathematicae*, 1977.

[Qui67] Daniel Quillen. Axiomatic homotopy theory. In *Homotopical Algebra*. Springer, 1967.

[Rey83] John Reynolds. Types, abstraction and parametric polymorphism. In *World Computer Congress*, 1983.

[Rob94] Edmund Robinson. Parametricity as isomorphism. *Theoretical Computer Science*, 1994.

[Tak01] Izumi Takeuti. The theory of parametricity in lambda cube (towards new interaction between category theory and proof theory). *Kyoto University Research Information Repository*, 2001.

[Tay99] Paul Taylor. *Practical foundations of mathematics*. Cambridge University Press, 1999.

[TTS21] Nicolas Tabareau, Éric Tanter, and Matthieu Sozeau. The marriage of univalence and parametricity. *Journal of the ACM*, 2021.

[Wad89] Philip Wadler. Theorems for free! In *International conference on Functional Programming Languages and Computer Architecture*, 1989.

APPENDIX A

Parametricity for categories with families respects equations

In this appendix, we give the computations proving that parametricity for categories with families respects equations. Section A.1 treats the case of unary parametricity for categories with families with product and unit types. Section A.2 extends this to arrow types and a universe.

A.1. With product and unit types

Our goal here is to check that given any equation:

$$s = t$$

in the theory of category with families from Section 2.6, the inductive definitions of s_* and t_* given in Definition 2.6.7 are equal in the theory of category with families.

First we check that equations for the calculus of substitutions are respected:

$$\begin{aligned} ((\sigma \circ \nu) \circ \delta)_* &= \sigma_*[\nu \circ \mathrm{w}, \nu_*][\delta \circ \mathrm{w}, \delta_*] \\ &= \sigma_*[\nu \circ \delta \circ \mathrm{w}, \nu_*[\delta \circ \mathrm{w}, \delta_*]] \\ &= (\sigma \circ (\nu \circ \delta))_* \end{aligned}$$

$$\begin{aligned} (\mathrm{id} \circ \sigma)_* &= \mathrm{v}[\mathrm{w}, \sigma_*] \\ &= \sigma_* \end{aligned}$$

$$\begin{aligned} (\sigma \circ \mathrm{id})_* &= \sigma_*[\mathrm{w}, \mathrm{v}] \\ &= \sigma_* \end{aligned}$$

For $\sigma : \mathrm{Hom}(\Gamma, \top)$, we have:

$$\begin{aligned} \epsilon_* &= \epsilon \\ &= \sigma_* \end{aligned}$$

$$\begin{aligned} \pi_1(\sigma, t)_* &= \pi_1(\sigma_*, t_*) \\ &= \sigma_* \end{aligned}$$

$$\begin{aligned} \pi_2(\sigma, t)_* &= \pi_2(\sigma_*, t_*) \\ &= t_* \end{aligned}$$

$$\begin{array}{rcl}(\pi_1(\sigma), \pi_2(\sigma))_* & = & (\pi_1(\sigma_*), \pi_2(\sigma_*)) \\ & = & \sigma_*\end{array}$$

$$\begin{array}{rcl}((\sigma, t) \circ \nu)_* & = & (\sigma_*, t_*)[\nu \circ \mathrm{w}, \nu_*] \\ & = & (\sigma_*[\nu \circ \mathrm{w}, \nu_*], t_*[\nu \circ \mathrm{w}, \nu_*]) \\ & = & (\sigma \circ \nu, t[\nu])_*\end{array}$$

We check that equations for unit types are respected:

$$\begin{array}{rcl}(\top[\sigma])_* & = & \top[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}] \\ & = & \top \\ & = & \top_*\end{array}$$

$$\begin{array}{rcl}(\epsilon[\sigma])_* & = & \mathrm{tt}[\sigma \circ \mathrm{w}, \sigma_*] \\ & = & \epsilon \\ & = & \epsilon_*\end{array}$$

For $t : \mathrm{Tm}(\Gamma, \top)$ we have:

$$\begin{array}{rcl}\epsilon_* & = & \epsilon \\ & = & t_*\end{array}$$

Now we check that equations for product types are respected:

$$\begin{array}{rl} & (\Sigma(A, B)[\sigma])_* \\ = & \Sigma(A_*[\mathrm{w}, \pi_1(\mathrm{v})], \\ & \quad B_*[\mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]])[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}] \\ = & \Sigma(A_*[\mathrm{w}, \pi_1(\mathrm{v})][\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}], \\ & \quad B_*[\mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]][\sigma \circ \mathrm{w}^3, \sigma_*[\mathrm{w}^2], \mathrm{v}[\mathrm{w}], \mathrm{v}]) \\ = & \Sigma(A_*[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \pi_1(\mathrm{v})], \\ & \quad B_*[\sigma \circ \mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\sigma_*[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]]) \\ = & \Sigma(A_*[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}][\mathrm{w}, \pi_1(\mathrm{v})], \\ & \quad B_*[\sigma \circ \mathrm{w}^3, \mathrm{v}[\mathrm{w}^2], (\sigma_*[\mathrm{w}^2, \pi_1(\mathrm{v})], \pi_2(\mathrm{v}))[\mathrm{w}], \mathrm{v}] \\ & \qquad [\mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]]) \\ = & \Sigma(A[\sigma], B[\sigma \circ \mathrm{w}, \mathrm{v}])_*\end{array}$$

$$\begin{array}{rcl}((s, t)[\sigma])_* & = & (s_*, t_*)[\sigma \circ \mathrm{w}, \sigma_*] \\ & = & (s_*[\sigma \circ \mathrm{w}, \sigma_*], t_*[\sigma \circ \mathrm{w}, \sigma_*]) \\ & = & (s[\sigma], t[\sigma])_*\end{array}$$

$$\begin{aligned}\pi_1(s,t)_* &= \pi_1(s_*,t_*)\\ &= s_*\end{aligned}$$

$$\begin{aligned}\pi_2(s,t)_* &= \pi_2(s_*,t_*)\\ &= t_*\end{aligned}$$

$$\begin{aligned}(\pi_1(t),\pi_2(t))_* &= (\pi_1(t_*),\pi_2(t_*))\\ &= t_*\end{aligned}$$

A.2. With arrow types and a universe

In Section 2.7 we defined arrow types and a universe. We prove that parametricity in Definition 2.7.4 respects their equations.

We check that equations for arrow types are respected:

$$\begin{aligned}
& (\Pi(A,B)[\sigma])_*\\
= {} & \Pi(A[\mathrm{w}^2],\Pi(A_*[\mathrm{w}^2,\mathrm{v}],\\
& \quad B_*[\mathrm{w}^4,\mathrm{v}[\mathrm{w}],(\mathrm{v}[\mathrm{w}^3],\mathrm{v}),\mathrm{ap}(\mathrm{v})[\mathrm{w}]]))[\sigma\circ\mathrm{w}^2,\sigma_*[\mathrm{w}],\mathrm{v}]\\
= {} & \Pi(A[\mathrm{w}^2][\sigma\circ\mathrm{w}^2,\sigma_*[\mathrm{w}],\mathrm{v}],\Pi(A_*[\mathrm{w}^2,\mathrm{v}][\sigma\circ\mathrm{w}^3,\sigma_*[\mathrm{w}^2],\mathrm{v}[\mathrm{w}],\mathrm{v}],\\
& \quad B_*[\mathrm{w}^4,\mathrm{v}[\mathrm{w}],(\mathrm{v}[\mathrm{w}^3],\mathrm{v}),\mathrm{ap}(\mathrm{v})[\mathrm{w}]][\sigma\circ\mathrm{w}^4,\sigma_*[\mathrm{w}^3],\mathrm{v}[\mathrm{w}^2],\mathrm{v}[\mathrm{w}],\mathrm{v}]))\\
= {} & \Pi(A[\sigma\circ\mathrm{w}^2],\Pi(A_*[\sigma\circ\mathrm{w}^3,\sigma_*[\mathrm{w}^2],\mathrm{v}],\\
& \quad B_*[\sigma\circ\mathrm{w}^4,\mathrm{v}[\mathrm{w}],(\sigma_*[\mathrm{w}^3],\mathrm{v}),\mathrm{ap}(\mathrm{v})[\mathrm{w}]]))\\
= {} & \Pi(A[\sigma\circ\mathrm{w}^2],\Pi(A_*[\sigma\circ\mathrm{w}^2,\sigma_*[\mathrm{w}],\mathrm{v}][\mathrm{w}^2,\mathrm{v}],\\
& \quad B_*[\sigma\circ\mathrm{w}^3,\mathrm{v}[\mathrm{w}^2],(\sigma_*[\mathrm{w}^2,\pi_1(\mathrm{v})],\pi_2(\mathrm{v})[\mathrm{w}],\mathrm{v}]\\
& \qquad [\mathrm{w}^4,\mathrm{v}[\mathrm{w}],(\mathrm{v}[\mathrm{w}^3],\mathrm{v}),\mathrm{ap}(\mathrm{v})[\mathrm{w}]]))\\
= {} & (\Pi(A[\sigma],B[\sigma\circ\mathrm{w},\mathrm{v}]))_*
\end{aligned}$$

$$\begin{aligned}
& (\lambda(t)[\sigma])_*\\
= {} & \lambda(\lambda(t_*[\mathrm{w}^3,\mathrm{v}[\mathrm{w}],(\mathrm{v}[\mathrm{w}^2],\mathrm{v})]))[\sigma\circ\mathrm{w},\sigma_*]\\
= {} & \lambda(\lambda(t_*[\mathrm{w}^3,\mathrm{v}[\mathrm{w}],(\mathrm{v}[\mathrm{w}^2],\mathrm{v})][\sigma\circ\mathrm{w}^3,\sigma_*[\mathrm{w}^2],\mathrm{v}[\mathrm{w}],\mathrm{v}]))\\
= {} & \lambda(\lambda(t_*[\sigma\circ\mathrm{w}^3,\mathrm{v}[\mathrm{w}],(\sigma_*[\mathrm{w}^2],\mathrm{v})])))\\
= {} & \lambda(\lambda(t_*[\sigma\circ\mathrm{w}^3,\mathrm{v}[\mathrm{w}],\sigma_*[\mathrm{w}^3,\mathrm{v}[\mathrm{w}^2],\mathrm{v}]))\\
= {} & \lambda(\lambda(t_*[\sigma\circ\mathrm{w}^2,\mathrm{v}[\mathrm{w}],\sigma_*[\mathrm{w}^2,\mathrm{v}.1,\pi_2(\mathrm{v})]][\mathrm{w}^3,\mathrm{v}[\mathrm{w}],(\mathrm{v}[\mathrm{w}^2],\mathrm{v})]))\\
= {} & \lambda(t[\sigma\circ\mathrm{w},\mathrm{v}])_*
\end{aligned}$$

$$\begin{aligned}
& \mathrm{ap}(\lambda(t))_* \\
= \; & \mathrm{ap}(\mathrm{ap}(\lambda(\lambda(t_*[\mathrm{w}^3, \mathrm{v}[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v})]))))[\mathrm{w}^2, \pi_1(\mathrm{v}), \mathrm{v}[\mathrm{w}], \pi_2(\mathrm{v})] \\
= \; & t_*[\mathrm{w}^3, \mathrm{v}[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v})][\mathrm{w}^2, \pi_1(\mathrm{v}), \mathrm{v}[\mathrm{w}], \pi_2(\mathrm{v})] \\
= \; & t_*[\mathrm{w}^2, \mathrm{v}[\mathrm{w}], (\pi_1(\mathrm{v}), \pi_2(\mathrm{v})] \\
= \; & t_*
\end{aligned}$$

$$\begin{aligned}
& \lambda(\mathrm{ap}(t))_* \\
= \; & \lambda(\lambda(\mathrm{ap}(\mathrm{ap}(t_*))[\mathrm{w}^2, \pi_1(\mathrm{v}), \mathrm{v}[\mathrm{w}], \pi_2(\mathrm{v})][\mathrm{w}^3, \mathrm{v}[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v})])) \\
= \; & \lambda(\lambda(\mathrm{ap}(\mathrm{ap}(t_*))[\mathrm{w}^3, \mathrm{v}[\mathrm{w}^2], \mathrm{v}[\mathrm{w}], \mathrm{v}])) \\
= \; & \lambda(\lambda(\mathrm{ap}(\mathrm{ap}(t_*)))) \\
= \; & t_*
\end{aligned}$$

Now we check that equations for the universe are respected:

$$\begin{aligned}
(\mathcal{U}[\sigma])_* &= \Pi(\mathrm{El}(\mathrm{v}), \mathcal{U})[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}] \\
&= \Pi(\mathrm{El}(\mathrm{v}), \mathcal{U}) \\
&= \mathcal{U}_*
\end{aligned}$$

$$\begin{aligned}
(\mathrm{El}(t)[\sigma])_* &= \mathrm{El}(\mathrm{ap}(t_*))[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}] \\
&= \mathrm{El}(\mathrm{ap}(t_*[\sigma \circ \mathrm{w}, \sigma_*])) \\
&= \mathrm{El}(t[\sigma])_*
\end{aligned}$$

$$\begin{aligned}
(\top_{\mathcal{U}}[\sigma])_* &= \lambda(\top_{\mathcal{U}})[\sigma \circ \mathrm{w}, \sigma_*] \\
&= \lambda(\top_{\mathcal{U}}) \\
&= (\top_{\mathcal{U}})_*
\end{aligned}$$

$$\begin{aligned}
& (\Sigma_{\mathcal{U}}(A, B)[\sigma])_* \\
= \; & \lambda(\Sigma_{\mathcal{U}}(\mathrm{ap}(A_*)[\mathrm{w}, \pi_1(\mathrm{v})], \\
& \qquad \mathrm{ap}(B_*)[\mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]]))[\sigma \circ \mathrm{w}, \sigma_*] \\
= \; & \lambda(\Sigma_{\mathcal{U}}(\mathrm{ap}(A_*)[\mathrm{w}, \pi_1(\mathrm{v})][\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}], \\
& \qquad \mathrm{ap}(B_*)[\mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]][\sigma \circ \mathrm{w}^3, \sigma_*[\mathrm{w}^2], \mathrm{v}[\mathrm{w}], \mathrm{v}])) \\
= \; & \lambda(\Sigma_{\mathcal{U}}(\mathrm{ap}(A_*)[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \pi_1(\mathrm{v})], \\
& \qquad \mathrm{ap}(B_*)[\sigma \circ \mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\sigma_*[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}])) \\
= \; & \lambda(\Sigma_{\mathcal{U}}(\mathrm{ap}(A_*)[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}][\mathrm{w}, \pi_1(\mathrm{v})], \\
& \qquad \mathrm{ap}(B_*)[\sigma \circ \mathrm{w}^3, \mathrm{v}[\mathrm{w}^2], (\sigma_*[\mathrm{w}^2, \pi_1(\mathrm{v})], \pi_2(\mathrm{v}))[\mathrm{w}], \mathrm{v}] \\
& \qquad\qquad [\mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]])) \\
= \; & \Sigma_{\mathcal{U}}(A[\sigma], B[\sigma \circ \mathrm{w}, \mathrm{v}])_*
\end{aligned}$$

$$
\begin{aligned}
& (\Pi_{\mathcal{U}}(A,B)[\sigma])_* \\
= \;& \lambda(\Pi_{\mathcal{U}}(A[\mathrm{w}^2], \Pi_{\mathcal{U}}(\mathrm{ap}(A_*)[\mathrm{w}^2,\mathrm{v}], \\
& \qquad \mathrm{ap}(B_*)[\mathrm{w}^4, \mathrm{v}[\mathrm{w}], (\mathrm{v}[\mathrm{w}^3],\mathrm{v}), \mathrm{ap}(\mathrm{v})[\mathrm{w}]])))[\sigma \circ \mathrm{w}, \sigma_*] \\
= \;& \lambda(\Pi_{\mathcal{U}}(A[\mathrm{w}^2][\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}], \\
& \qquad \Pi_{\mathcal{U}}(\mathrm{ap}(A_*)[\mathrm{w}^2,\mathrm{v}][\sigma \circ \mathrm{w}^3, \sigma_*[\mathrm{w}^2], \mathrm{v}[\mathrm{w}], \mathrm{v}], \\
& \qquad\quad \mathrm{ap}(B_*)[\mathrm{w}^4, \mathrm{v}[\mathrm{w}], (\mathrm{v}[\mathrm{w}^3],\mathrm{v}), \mathrm{ap}(\mathrm{v})[\mathrm{w}]] \\
& \qquad\qquad [\sigma \circ \mathrm{w}^4, \sigma_*[\mathrm{w}^3], \mathrm{v}[\mathrm{w}^2], \mathrm{v}[\mathrm{w}], \mathrm{v}]))) \\
= \;& \lambda(\Pi_{\mathcal{U}}(A[\sigma \circ \mathrm{w}^2], \Pi_{\mathcal{U}}(\mathrm{ap}(A_*)[\sigma \circ \mathrm{w}^3, \sigma_*[\mathrm{w}^2], \mathrm{v}], \\
& \qquad \mathrm{ap}(B_*)[\sigma \circ \mathrm{w}^4, \mathrm{v}[\mathrm{w}], (\sigma_*[\mathrm{w}^3],\mathrm{v}), \mathrm{ap}(\mathrm{v})[\mathrm{w}]]))) \\
= \;& \lambda(\Pi_{\mathcal{U}}(A[\sigma \circ \mathrm{w}^2], \Pi_{\mathcal{U}}(\mathrm{ap}(A_*)[\sigma \circ \mathrm{w}^2, \sigma_*[\mathrm{w}], \mathrm{v}][\mathrm{w}^2, \mathrm{v}], \\
& \qquad \mathrm{ap}(B_*)[\sigma \circ \mathrm{w}^3, \mathrm{v}[\mathrm{w}^2], (\sigma_*[\mathrm{w}^2, \mathrm{v}.1], \pi_2(\mathrm{v}))[\mathrm{w}], \mathrm{v}] \\
& \qquad\quad [\mathrm{w}^4, \mathrm{v}[\mathrm{w}], (\mathrm{v}[\mathrm{w}^3],\mathrm{v}), (\mathrm{ap}(\mathrm{v}))[\mathrm{w}]]))) \\
= \;& \Pi_{\mathcal{U}}(A[\sigma], B[\sigma \circ \mathrm{w}, \mathrm{v}])_*
\end{aligned}
$$

Finally we check that equations for El are respected:

$$
\begin{aligned}
\mathrm{El}(\top_{\mathcal{U}})_* &= \mathrm{El}(\mathrm{ap}(\lambda(\top_{\mathcal{U}}))) \\
&= \top \\
&= \top_*
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{El}(\Sigma_{\mathcal{U}}(A,B))_* &= \mathrm{El}(\mathrm{ap}(\lambda(\Sigma_{\mathcal{U}}(\mathrm{ap}(A_*)[\eta_1], \mathrm{ap}(B_*)[\eta_2])))) \\
&= \Sigma(\mathrm{El}(\mathrm{ap}(A_*))[\eta_1], \mathrm{El}(\mathrm{ap}(B_*))[\eta_2]) \\
&= \Sigma(\mathrm{El}(A), \mathrm{El}(B))_*
\end{aligned}
$$

$$
\begin{aligned}
& \mathrm{El}(\Pi_{\mathcal{U}}(A,B))_* \\
= \;& \mathrm{El}(\mathrm{ap}(\lambda(\Pi_{\mathcal{U}}(A[\sigma_1], \Pi_{\mathcal{U}}(\mathrm{ap}(A_*)[\sigma_2], \mathrm{ap}(B_*)[\sigma_3]))))) \\
= \;& \Pi(\mathrm{El}(A)[\sigma_1], \Pi(\mathrm{El}(\mathrm{ap}(A_*))[\sigma_2], \mathrm{El}(\mathrm{ap}(B_*))[\sigma_3])) \\
= \;& \Pi(\mathrm{El}(A), \mathrm{El}(B))_*
\end{aligned}
$$

where:

$$
\begin{aligned}
\eta_1 &= (\mathrm{w}, \pi_1(\mathrm{v})) \\
\eta_2 &= (\mathrm{w}^3, \pi_1(\mathrm{v})[\mathrm{w}], (\mathrm{v}[\mathrm{w}^2], \mathrm{v}), \pi_2(\mathrm{v})[\mathrm{w}]) \\
\sigma_1 &= \mathrm{w}^2 \\
\sigma_2 &= (\mathrm{w}^2, \mathrm{v}) \\
\sigma_3 &= (\mathrm{w}^4, \mathrm{v}[\mathrm{w}], (\mathrm{v}[\mathrm{w}^3], \mathrm{v}), (\mathrm{ap}(\mathrm{v}))[\mathrm{w}])
\end{aligned}
$$

APPENDIX B

Quotient inductive-inductive types

This appendix sketches the inductive definition of displayed algebras and their sections as given in [**KKA19**] and [**KK20**]. It is not intended as a rigorous presentation, but as a help in navigating these technical articles.

B.1. Displayed algebras

We mentioned in Proposition 2.2.9 that for any signature Γ, we can define inductively a type:

$$\mathrm{Disp}_\Gamma \quad : \quad \mathrm{Ty}(\Gamma) \tag{B.1.1}$$

such that we have:

$$\begin{tikzcd}
\mathrm{Alg}_{\Gamma,\mathrm{Disp}_\Gamma} \arrow[rr, "\simeq"] \arrow[dr, "U"'] & & \mathrm{Alg}_\Gamma^{\to} \arrow[dl, "\mathrm{cod}"] \\
& \mathrm{Alg}_\Gamma &
\end{tikzcd}$$

In order to define Disp_Γ inductively on Γ, we need to define by simultaneous induction:

$$\begin{aligned}
\mathrm{Disp}_\Gamma \quad &: \quad \mathrm{Ty}(\Gamma) && \text{(B.1.2)}\\
\mathrm{Disp}_A \quad &: \quad \mathrm{Ty}(\Gamma, \mathrm{Disp}_\Gamma, A) && \text{(B.1.3)}\\
\mathrm{Disp}_\sigma \quad &: \quad \mathrm{Tm}((\Gamma, \mathrm{Disp}_\Gamma), \mathrm{Disp}_\Delta[\sigma \circ \mathrm{w}]) && \text{(B.1.4)}\\
\mathrm{Disp}_a \quad &: \quad \mathrm{Tm}((\Gamma, \mathrm{Disp}_\Gamma), \mathrm{Disp}_A[\mathrm{id}, a]) && \text{(B.1.5)}
\end{aligned}$$

It turns out that we can simply use unary parametricity from Definitions 2.6.7 and 2.7.4. So Disp_Γ is defined as Γ_*. The only new types in signatures are extensional identity types:

- We define:

  $$\mathrm{Disp}_{\mathrm{Id}(s,t)} \quad = \quad \mathrm{Id}(\mathrm{Disp}_s, \mathrm{Disp}_t) \tag{B.1.6}$$

  This is well-typed because $\mathrm{Id}(s, t)$ implies $s = t$, so that Disp_s and Disp_t have the same type.
- In order to prove that the rule:

  $$\mathrm{refl} \quad : \quad s = t \longrightarrow \mathrm{Tm}(\Gamma, \mathrm{Id}(s, t)) \tag{B.1.7}$$

  is respected by parametricity, we need to define $\mathrm{Disp}_{\mathrm{refl}}$. To do this we assume $s = t$ and give an inhabitant in:

  $$\mathrm{Disp}_{\mathrm{Id}(s,t)} \quad = \quad \mathrm{Id}(\mathrm{Disp}_s, \mathrm{Disp}_t) \tag{B.1.8}$$

  This holds by reflexivity, as $s = t$ implies $\mathrm{Disp}_s = \mathrm{Disp}_t$.

- Now we need to prove that the rule:
$$\mathrm{ext} \quad : \quad \mathrm{Tm}(\Gamma, \mathrm{Id}(s,t)) \to s = t \tag{B.1.9}$$
is respected by parametricity. To do this we need to prove assuming:
$$p \quad : \quad \mathrm{Id}(s,t) \tag{B.1.10}$$
that:
$$\mathrm{Disp}_s \quad = \quad \mathrm{Disp}_t \tag{B.1.11}$$
But we have:
$$\mathrm{Disp}_p \quad : \quad \mathrm{Id}(\mathrm{Disp}_s, \mathrm{Disp}_t) \tag{B.1.12}$$
and we can conclude by extensionality.

Now we give a few interesting examples of signatures and explain how their displayed algebras relate to maps to an algebra:

- Consider the signature $X : \mathcal{U}$. Then we have:
$$\mathrm{Disp}_{X:\mathcal{U}} \quad = \quad X \to \mathcal{U} \tag{B.1.13}$$
The equivalence:
$$\mathrm{Alg}_{X:\mathcal{U}, X\to\mathcal{U}} \quad \simeq \quad \mathrm{Alg}^{\to}_{X:\mathcal{U}} \tag{B.1.14}$$
depends on the meta-theory. It holds for sets. It sends a family:
$$(X, \widetilde{X}) \quad : \quad \mathrm{Alg}_{X:\mathcal{U},\, X\to\mathcal{U}} \tag{B.1.15}$$
to the map:
$$\pi_1 \quad : \quad (x : X) \times \widetilde{X}(x) \to X \tag{B.1.16}$$
This is indeed an object in $\mathrm{Alg}^{\to}_{X:\mathcal{U}}$.
- Consider the signature:
$$\Gamma \quad = \quad (X : \mathcal{U},\ x : X) \tag{B.1.17}$$
Then we have:
$$\mathrm{Disp}_\Gamma \quad = \quad (\widetilde{X} : X \to \mathcal{U}) \times \widetilde{X}(x) \tag{B.1.18}$$
Given:
$$(X, x, \widetilde{X}, \widetilde{x}) : \mathrm{Alg}_{\Gamma, \mathrm{Disp}_\Gamma} \tag{B.1.19}$$
we have:
$$(x, \widetilde{x}) \quad : \quad (x : X) \times \widetilde{X}(x) \tag{B.1.20}$$
$$\pi_1(x, \widetilde{x}) \quad = \quad x \tag{B.1.21}$$
So that π_1 is an object in $\mathrm{Alg}^{\to}_\Gamma$.
- Consider the signature:
$$\Delta \quad = \quad (X, Y : \mathcal{U},\ f : X \to Y) \tag{B.1.22}$$
Then we have that Disp_Δ is the type:
$$(\widetilde{X} : X \to \mathcal{U}) \times (\widetilde{Y} : Y \to \mathcal{U}) \times \big((x : X) \to \widetilde{X}(x) \to \widetilde{Y}(f(x))\big) \tag{B.1.23}$$
Given:
$$(X, Y, f, \widetilde{X}, \widetilde{Y}, \widetilde{f}) \quad : \quad \mathrm{Alg}_{\Delta, \mathrm{Disp}_\Delta} \tag{B.1.24}$$

we have a commutative square:

$$\begin{tikzcd}
(x : X) \times \widetilde{X}(x) \arrow[r, "{(x,\widetilde{x}) \mapsto (f(x),\widetilde{f}(x,\widetilde{x}))}"] \arrow[d, "\pi_1"'] & (y : Y) \times \widetilde{Y}(y) \arrow[d, "\pi_1"] \\
X \arrow[r, "f"'] & Y
\end{tikzcd}$$

This is indeed an object in $\mathrm{Alg}_{\Delta}^{\rightarrow}$.

- Consider the signature:

$$\Theta \quad = \quad (X : \mathfrak{U},\ x, y : X,\ \mathrm{Id}(x, y)) \tag{B.1.25}$$

Then Disp_{Θ} is the type:

$$(\widetilde{X} : X \to \mathfrak{U}) \times (\widetilde{x} : \widetilde{X}(x)) \times (\widetilde{y} : \widetilde{X}(y)) \times \mathrm{Id}(\widetilde{x}, \widetilde{y}) \tag{B.1.26}$$

Here $\mathrm{Id}(\widetilde{x}, \widetilde{y})$ is well-typed because we assumed a variable in $\mathrm{Id}(x, y)$, so that by extensionnality $x = y$ and:

$$\widetilde{X}(x) \quad = \quad \widetilde{X}(y) \tag{B.1.27}$$

We see that assuming:

$$(X, x, y, p, \widetilde{X}, \widetilde{x}, \widetilde{y}, \widetilde{p}) \quad : \quad \mathrm{Alg}_{\Theta, \mathrm{Disp}_{\Theta}} \tag{B.1.28}$$

we have that:

$$\widetilde{p} \quad : \quad \mathrm{Id}(\widetilde{x}, \widetilde{y}) \tag{B.1.29}$$

implies that $\widetilde{x} = \widetilde{y}$ by extensionality, so that:

$$(x, \widetilde{x}) \quad = \quad (y, \widetilde{y}) \tag{B.1.30}$$

Then we have an object in $\mathrm{Alg}_{\Theta}^{\rightarrow}$.

B.2. Sections of a displayed algebra

Now we give a similar overview for sections. This means that we need to define a type:

$$\mathrm{Sec}_{\Gamma} \quad : \quad \mathrm{Ty}(\Gamma, \mathrm{Disp}_{\Gamma}) \tag{B.2.1}$$

inductively on Γ such that:

$$\begin{tikzcd}
\mathrm{Alg}_{\Gamma, \mathrm{Disp}_{\Gamma}, \mathrm{Sec}_{\Gamma}} \arrow[r, "\simeq"] \arrow[d] & \mathrm{Alg}_{\Gamma}^{s} \arrow[d] \\
\mathrm{Alg}_{\Gamma, \mathrm{Disp}_{\Gamma}} \arrow[r, "\simeq"'] & \mathrm{Alg}_{\Gamma}^{\rightarrow}
\end{tikzcd}$$

where $\mathrm{Alg}_{\Gamma}^{s}$ is the category of arrows with a section in Alg_{Γ}. Morphisms in this category are required to commute with sections.

In order to define Sec_{Γ} inductively on Γ, we need to define by simultaneous induction:

$$\mathrm{Sec}_{\Gamma} \quad : \quad \mathrm{Ty}(\Gamma, \mathrm{Disp}_{\Gamma}) \tag{B.2.2}$$

$$\mathrm{Sec}_{A} \quad : \quad \mathrm{Ty}(\Gamma, \mathrm{Disp}_{\Gamma}, \mathrm{Sec}_{\Gamma}, A, \mathrm{Disp}_{A}) \tag{B.2.3}$$

$$\mathrm{Sec}_{\sigma} \quad : \quad \mathrm{Tm}((\Gamma, \mathrm{Disp}_{\Gamma}, \mathrm{Sec}_{\Gamma}), \mathrm{Sec}_{\Delta}[\sigma, \mathrm{Disp}_{\sigma}]) \tag{B.2.4}$$

$$\mathrm{Sec}_{a} \quad : \quad \mathrm{Tm}((\Gamma, \mathrm{Disp}_{\Gamma}, \mathrm{Sec}_{\Gamma}), \mathrm{Sec}_{A}[\mathrm{id}, a, \mathrm{Disp}_{a}]) \tag{B.2.5}$$

The definitions for unit, product and arrow types are very similar to parametricity. We give the definitions for the universe and extensional identity types:

- For the universe we define:

$$\mathrm{Sec}_{\mathcal{U}}[X : \mathcal{U}, \widetilde{X} : X \to \mathcal{U}] \quad = \quad (x : X) \to \widetilde{X}(x) \tag{B.2.6}$$
$$\mathrm{Sec}_{\mathrm{El}(A)}[x : \mathrm{El}(A), \widetilde{x} : \mathrm{El}(\widetilde{A}(x))] \quad = \quad \mathrm{Id}(\mathrm{Sec}_A(x), \widetilde{x}) \tag{B.2.7}$$

- For identity types we just define:

$$\mathrm{Sec}_{\mathrm{Id}(x,y)} \quad = \quad \top \tag{B.2.8}$$

  This means that $\mathrm{Sec}_{\mathrm{refl}}$ is an inhabitant of $\top$. For extensionality, we need to check that assuming:

$$p \quad : \quad \mathrm{Id}_A(x, y) \tag{B.2.9}$$

  we have:

$$\mathrm{Sec}_x \quad = \quad \mathrm{Sec}_y \tag{B.2.10}$$

  But since $A : \mathcal{U}$ we have that Sec_A is an identity type, and we can conclude by unicity of identity proofs.

  Remark B.2.1. If we used intentional identity types as in [**KK20**], then $\mathrm{Sec}_{\mathrm{Id}_A(x,y)}$ would require that Sec_A sends:

$$p \quad : \quad \mathrm{Id}(x, y) \tag{B.2.11}$$

  to $\widetilde{p}$ in the appropriate sense. We do not have to require anything in our case by unicity of identity proofs.

Now we give a few interesting examples, where we check that sections of a displayed algebra indeed give sections of the corresponding map:

- Consider the signature $X : \mathcal{U}$, then we have:

$$\mathrm{Sec}_{X:\mathcal{U}} \quad = \quad (x : X) \to \widetilde{X}(x) \tag{B.2.12}$$

  Assume given:

$$(X, \widetilde{X}, s_X) \quad : \quad \mathrm{Alg}_{X:\mathcal{U}, \mathrm{Disp}_{X:\mathcal{U}}, \mathrm{Sec}_{X:\mathcal{U}}} \tag{B.2.13}$$

  then we have:

$$\begin{tikzcd} (x : X) \times \widetilde{X}(x) \arrow[rr, "\mathrm{id}"] \arrow[dr, "\pi_1"'] & & (x : X) \times \widetilde{X}(x) \\ & X \arrow[ur, "x \mapsto (x, s_X(x))"'] & \end{tikzcd}$$

  Giving an object in $\mathrm{Alg}^s_{X:\mathcal{U}}$.

- Consider the signature:

$$\Gamma \quad = \quad (X : \mathcal{U},\ x : X) \tag{B.2.14}$$

  Then we have:

$$\mathrm{Sec}_\Gamma \quad = \quad (s_X : (x : X) \to \widetilde{X}(x)) \times \mathrm{Id}(s_X(x), \widetilde{x}) \tag{B.2.15}$$

  Assume given:

$$(X, x, \widetilde{X}, \widetilde{x}, s_X, p) \quad : \quad \mathrm{Alg}_{\Gamma, \mathrm{Disp}_\Gamma, \mathrm{Sec}_\Gamma} \tag{B.2.16}$$

where:

$$p \quad : \quad \mathrm{Id}(s_X(x), \widetilde{x}) \tag{B.2.17}$$

Then the section:

$$\lambda(x : X).\,(x, s_X(x)) \quad : \quad X \to (x : X) \times \widetilde{X}(x) \tag{B.2.18}$$

sends x to:

$$(x, \widetilde{x}) \quad : \quad (x : X) \times \tilde{X}(x) \tag{B.2.19}$$

So we have an object in Alg^s_Γ.

- Consider the signature:

$$\Delta \quad = \quad (X, Y : \mathcal{U},\ f : X \to Y) \tag{B.2.20}$$

Then we have that Sec_Δ is the type:

$$\begin{aligned} &(s_X : (x : X) \to \widetilde{X}(x)) \\ \times\quad &(s_Y : (y : Y) \to \widetilde{Y}(y)) \\ \times\quad &(x : X) \to \mathrm{Id}(s_Y(f(x)), \widetilde{f}(x, s_X(x))) \end{aligned} \tag{B.2.21}$$

Given:

$$(X, Y, f, \widetilde{X}, \widetilde{Y}, \widetilde{f}, s_X, s_Y, p) \quad : \quad \mathrm{Alg}_{\Delta, \mathrm{Disp}_\Delta, \mathrm{Sec}_\Delta} \tag{B.2.22}$$

where:

$$p(x) \quad : \quad \mathrm{Id}(s_Y(f(x)), \widetilde{f}(x, s_X(x))) \tag{B.2.23}$$

We have a commutative square:

$$\begin{tikzcd} (x : X) \times \widetilde{X}(x) \arrow[r, "(x,\widetilde{x}) \mapsto (f(x), \widetilde{f}(x,\widetilde{x}))"] & (y : Y) \times \widetilde{Y}(y) \\ X \arrow[u, "x \mapsto (x, s_X(x))"] \arrow[r, "f"'] & Y \arrow[u, "y \mapsto (y, s_Y(y))"'] \end{tikzcd}$$

So we have an object in Alg^s_Δ.

- Consider the signature:

$$\Theta \quad = \quad (X : \mathcal{U},\ x, y : X,\ \mathrm{Id}(x, y)) \tag{B.2.24}$$

Then Sec_Θ is the type:

$$(s_X : (x : X) \to \widetilde{X}(x)) \times \mathrm{Id}(s_X(x), \widetilde{x}) \times \mathrm{Id}(s_X(y), \widetilde{y}) \tag{B.2.25}$$

We see that assuming:

$$(X, x, y, p, \widetilde{X}, \widetilde{x}, \widetilde{y}, \widetilde{p}, s_X, p_x, p_y) : \mathrm{Alg}_{\Theta, \mathrm{Disp}_\Theta, \mathrm{Sec}_\Theta} \tag{B.2.26}$$

where:

$$p_x \quad : \quad \mathrm{Id}(s_X(x), \widetilde{x}) \tag{B.2.27}$$

$$p_y \quad : \quad \mathrm{Id}(s_X(y), \widetilde{y}) \tag{B.2.28}$$

we have that the section:

$$\lambda(x : X).\,(x, s_X(x)) \quad : \quad X \to (x : X) \times \widetilde{X}(x) \tag{B.2.29}$$

sends x to $(x, \widetilde{x})$ and y to $(y, \widetilde{y})$. So we indeed have an object in Alg^s_Θ.