Notions of parametricity as monoidal models for type theory

Hugo Moeneclaey Université de Paris, Inria Paris, CNRS, IRIF, France

September 5, 2022

Abstract

This article gives a solid theoretical grounding to the observation that cubical structures arise naturally when working with parametricity. Following [17], we claim that cubical models are cofreely parametric.

We use categories, lex categories or clans as models of type theory. In this context we define notions of parametricity as monoidal models, and parametric models as modules. This covers not only the usual parametricity where any type comes with a relation, but also variants where it comes with a predicate, a reflexive relation, two relations, and many more.

In this setting we prove that forgetful functors from parametric models to arbitrary ones have left and right adjoints. Moreover we give explicit compact descriptions for these freely and cofreely parametric models.

Then we give many examples of notion of parametricity, allowing to build the following as cofreely parametric models:

- Categories of cubical objects for any variant of cube.
- Lex categories of truncated semi-cubical (or cubical with reflexivities only) objects.
- Clans of Reedy fibrant semi-cubical (or cubical with reflexivities only) objects.

Contents

1	Parametricity for categories		
	1.1	A first definition	5
	1.2	Reformulation using semi-cubes	7
2	An	axiomatisation for this situation	9
2	An 2.1	axiomatisation for this situation Notions of parametricity and parametric objects	9 9

3	Parametricity for left exact categories		14
	3.1	The category of lex categories is symmetric monoidal closed	15
	3.2	Notions of parametricity for lex categories	19
4	Par	ametricity for clans	24
	4.1	Clans with strictly commuting limits	25
	4.2	Monoidal structure on clans	28
	4.3	Notions of parametricity for clans	37

Overview

Introduction to type theory

Type theories are foundational systems for constructive mathematics [16]. A type theory is given by its rules, which allow to build types (corresponding to sets and propositions) and terms in types (corresponding to elements in sets and proofs of propositions).

In this article we adopt a semantical point view [10, 11]. This means that we will study models of type theory. Such a model consists of notions of types and terms obeying the rules of the given type theory. From this point of view the syntax of type theory constitutes the simplest model, technically the initial model.

Many definitions of model have been proposed in the literature, but many of them have two important common features:

- They are generalised algebraic notions [8]. This means that we have a natural notion of morphism between models, forming a locally presentable category [1]. These categories are well-behaved, for example all forgetful functors between locally presentable categories always have left adjoints.
- A model of type theory has an underlying category. This allows to translate many categorical intuitions and techniques to models of type theory.

It is not uncommon to ask for enough structure in a model of type theory so that it can reasonably be considered as a place where all (constructive) mathematics can happen. In this article we will not consider so much structure in models, focusing on modeling types and dependencies between them.

Introduction to parametricity and cubes

Parametricity was first introduced as a syntactic propriety of system F [19]. It can be summarised as the observation that any type comes with a relation and any term preserves these relations. This is proved by mutual induction on types and terms.

Intuitively, this means that terms in system F treat their inputs uniformly. For example it implies that any term of type $X \to X$ with X a type variable is extensionally equal to the identity function [22]. This techniques has been extended from system F to type theory [4, 14, 21].

We will consider parametric models, which are models with:

- For each type A a relation A_* over A.
- For each term t a new term t_* witnessing that t preserves the relevant relations.

Moreover A_* and t_* should obey the equations inductively defining parametricity. It is a known phenomena that when attempting to build parametric models, cubical structures arise:

- A model for (a variant of) parametricity was given by Bernardy, Coquand and Moulin using (a variant of) cubical sets [3].
- Johann and Sojakova defined so-called *higher-dimensional parametricity* using cubes in [12],
- Cavallo and Harper gave a cubical type theory (i.e. a type theory inspired by cubical models) supporting some kind of parametricity [9].

In this article we will explain and illustrate this phenomena.

Content

Many variants of parametricity can be considered, where any type comes with something else than a relation. In this article we show that for many such variant of parametricity we have a string of adjoint functors:



as in [17]. We provide a few notions of parametricity with their corresponding cubical structure, to give a bit of intuition:

Parametricty	Relation	Semi-cubes
Internal parametricity	Reflexive relations	Cubes
Unary parametricity	Predicate	Augmented semi-simplices
Bi -parametricity	Two relations	Semi-bicubes

We will consider three notions of model for type theory:

• Plain categories, without any extra structure. This is usually not considered as a model of type theory. Nevertheless we believe that this example is crucial in providing an introduction to parametricity for categorically-minded readers.

- Left exact categories (lex categories for short). They are particularly simple models for type theory with Σ-types and a unit type.
- Clans [13], generalising lex categories. They model type theory with Σ -types and a unit, where a type A over Γ is fully determined by its weakening:

$$(\Gamma, A) \to \Gamma \tag{1}$$

A clan is a category with a class of map called fibration obeying a few of the axioms of model categories, so they will not look too unfamiliar to homotopically-minded reader.

We give an early disclaimer on conceivable extensions:

- We do not foresee any trouble when extending the consideration presented here to categories with families, comprehension categories and similar notions of models of type theory.
- We conjecture that extensions to models with more inductive types are fine, provided that we have enough inductive types to define all the parametricity relations involved.
- Having II-types and universes lead to serious trouble, and we do not attempt to treat them here.

Now we give a precise plan:

- In Section 1 we define the usual parametricity (where every type comes with a relation) for categories. Motivated by a reformulation of this definition, we define a notion of parametricity as a monoidal category, and a parametric model as a module. We see that the monoidal category correponding to the usual parametricity is the (opposite of the) category of semi-cubes.
- In Section 2 we axiomatise the situation from Section 1, by asking for a symmetric monoidal closed category of models. Then we define a notion of parametricity as a monoid in this category (that is a model of type theory with a compatible monoidal structure on its underlying category). Moreover we define parametric models as modules, and we give compact descriptions for the left and right adjoints to the functor forgetting the module structure.
- In Section 3 we present notions of parametricity for lex categories as monoids for a symmetric monoidal closed structure on the category of lex categories. We show that notions of parametricity for categories can be extended to lex categories. Moreover we show that non-iterated parametricities can be defined for lex categories, giving rise to truncated semi-cubical (or cubical with reflexivities only) objects, for example graphs.

• In Section 4 we present notions of parametricity for clans. We show how notions of parametricity for categories can be extended to clans, giving rise to various cubical objects with pointwise fibrations. Moreover we show that clans of semi-cubical (or cubical with reflexivities only) objects with Reedy fibrations are cofreely parametric for suitable notions of parametricity.

1 Parametricity for categories

This preliminary section is meant to motivate the abstract axiomatisation of the next section, by explaining the reasoning leading to it. We consider plain categories as models for type theory, and examine parametricity for them very closely.

1.1 A first definition

Recall the most basic form of parametricity is when any type comes with a relation, and any term preserves these relations. First we translate this as directly as possible to a category.

Definition 1. A category C is parametric if we are given:

• An endofunctor:

$$\underline{}_*: \mathcal{C} \to \mathcal{C} \tag{2}$$

• For any $X : \mathcal{C}$ two morphisms:

$$d_X^0, d_X^1: X_* \to X \tag{3}$$

natural in X.

Remark 2. This definition reflects our intuition on parametricity. Indeed:

• For Xa: C the data of:

$$d_X^0, d_X^1: X_* \to X \tag{4}$$

is a relation over X internal to C.

 The fact that _* is a functor with d⁰ and d¹ natural precisely means that morphisms in C preserve the relations.

Given an object X in a parametric category C, we can iterate $__*$, building the following diagram:

$$X \underbrace{\stackrel{d_{X}^{0}}{=}}_{d_{X}^{1}} X_{*} \underbrace{\stackrel{(d_{X}^{0})_{*}}{\stackrel{d_{X}^{0}}{=}}}_{(d_{X}^{1})_{*}} X_{**} \cdots$$

This will turn out to be a semi-cubical objects in C with X as its object of points. See Remark 14 for more details.

Remark 3. A notion of parametricity where $__*$ can be applied as many times as we want is called iterated. We will tell more about non-iterated (also called truncated) notions of parametricity later. The most basic example is when $__*$ can only be applied once, so that X_{**} does not give any new information on X.

Example 4. We give examples of parametric categories.

• We can define:

$$X_* = X \tag{5}$$

$$d_X^0(x) = x \tag{6}$$

 $d_X^1(x) = x \tag{7}$

Any type comes with its equality relation. This works in any category.

• If the category has products, we can consider:

$$X_* = X \times X \tag{8}$$

$$d_X^0(x,y) = x \tag{9}$$

$$l_X^1(x,y) = y \tag{10}$$

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

• If we have an initial object \perp we can define:

$$X_* = \bot \tag{11}$$

Any type comes with the empty (i.e. always false) relation.

In all these examples the maps:

$$(d_X^0, d_X^1): X_* \to X \times X \tag{12}$$

are monomorphisms, so that their fibers are sub-singletons. Assuming the law of excluded middle, there is no other such notions of parametricity on the category of sets.

Example 5. We give examples where the maps:

$$(d_X^0, d_X^1) : X_* \to X \times X \tag{13}$$

are not monomorphisms.

• In any cartesian closed category of spaces containing the unit interval [0,1] in \mathbb{R} , we can define:

$$X_* = X^{[0,1]} \tag{14}$$

$$d_X^0(p) = p(0) \tag{15}$$

$$d_X^1(p) = p(1) (16)$$

For any points x and y in X, the fiber X_* over (x, y) is the space of paths from x to y in X.

The cubical space constructed from any space X by iterating $__*$ is the usual cubical nerve of X with its natural topology.

• This can be generalised to any cartesian closed category with an object I and two maps:

$$0, 1: \top \to I \tag{17}$$

by defining:

$$X_* = X^I \tag{18}$$

$$d_X^0 = X^I \xrightarrow{X^0} X^\top \cong X \tag{19}$$

$$d_X^1 = X^I \xrightarrow{X^1} X^\top \cong X \tag{20}$$

• Assuming a functorial factorisation system in a category C, we can define a parametricity for C by factoring diagonals as follows:

$$X \longrightarrow X_* \xrightarrow{(d_X^0, d_X^1)} X \times X$$

1.2 Reformulation using semi-cubes

Recall the following description for category of semi-cubes:

Definition 6. Let \Box is the free strict monoidal category generated by:

- An object \mathbb{I} .
- Two morphisms:

$$d^0, d^1: \mathbb{I} \to 1 \tag{21}$$

where 1 is the monoidal unit.

The reader unfamiliar with semi-cubical objects can take the following as a definition.

Proposition 7. A functor from \Box to C is a semi-cubical object in C.

Remark 8. It is standard to define semi-cubical objects as presheaves on \Box^{op} , but we do not want to include opposites in our axiomatisation of the situation. We will give remarks whenever this could lead to confusion.

Remark 9. Objects in \Box are of the form $\mathbb{I} \otimes \cdots \otimes \mathbb{I}$ and morphisms are tensors of d^0 , d^1 and $id_{\mathbb{I}}$.

For example morphisms going from $\mathbb{I} \otimes \mathbb{I} \otimes \mathbb{I}$ to $\mathbb{I} \otimes \mathbb{I}$ are:

$$d^{\epsilon} \otimes \mathbb{I} \otimes \mathbb{I} \tag{22}$$

$$\mathbb{I} \otimes d^{\epsilon} \otimes \mathbb{I} \tag{23}$$

$$\mathbb{I} \otimes \mathbb{I} \otimes d^{\epsilon} \tag{24}$$

where $\epsilon = 0, 1$.

There is a clear analogy between the natural transformations:

$$d^0, d^1 : __* \to \mathrm{Id} \tag{25}$$

and the generators for semi-cubes:

$$d^0, d^1: \mathbb{I} \to 1 \tag{26}$$

To make this precise we need auxiliary definitions. The first one is very well-known:

Definition 10. For C a category, we write End_{C} for the strict monoidal category of endofunctors of C, with composition as tensor and identity as unit.

The second definition is not as common, but elementary.

Definition 11. Let \mathcal{M} be a strict monoidal category. An \mathcal{M} -module consists of:

- A category C.
- A strict monoidal functor from \mathcal{M} to $\operatorname{End}_{\mathcal{C}}$.

This is just the usual definition of an action of monoid, valid in any cartesian closed category, specialised to the category of categories.

Remark 12. We could equivalently define an \mathcal{M} -module as a category \mathcal{C} with a functor:

$$\underline{} \otimes \underline{} : \mathcal{M} \times \mathcal{C} \to \mathcal{C} \tag{27}$$

such that:

$$(i \otimes j) \otimes X = i \otimes (j \otimes X) \tag{28}$$

$$1 \otimes X = X \tag{29}$$

functorially in i, j and X.

Now we can state:

Lemma 13. Being parametric is the same as being a \Box -module.

Proof. By Proposition 7, giving a monoidal functor from \Box to \mathcal{D} is equivalent to giving an object $X : \mathcal{D}$ with two morphisms from X to 1. So a monoidal functor from \Box to $\text{End}_{\mathcal{C}}$ corresponds precisely to $__*$ an endofunctor of \mathcal{C} with two natural transformations from $__*$ to the identity endofunctor. \Box

Remark 14. For any \Box -module C with an object X, we have a functor:

$$F : \Box \to \mathcal{C} \tag{30}$$

$$F(i) = i \otimes X \tag{31}$$

such that F(1) = X, so that X is indeed the object of points for a semi-cubical object in C.

2 An axiomatisation for this situation

Now, using insight from the first section, we will define an abstract notion of parametricity. We will use the example of categories throughout to give intuitions on definitions and results presented here.

Notation 15. Everywhere in this section we will assume \mathcal{U} a symmetric monoidal closed category, with:

- A product $_ \otimes _$.
- A unit 1.
- A linear arrow $_ \multimap _$.

An object in \mathcal{U} is seen as a model of type theory.

Example 16. We can take the category of categories as \mathcal{U} . Its monoidal closed structure is in fact even cartesian.

Example 17. We can take the category of abelian groups as \mathcal{U} . The tensor is genuinely linear in the sense that it is not a cartesian product.

2.1 Notions of parametricity and parametric objects

Definition 18. A notion of parametricity for \mathcal{U} is a monoid in \mathcal{U} . This means it consists of $\mathcal{M}: \mathcal{U}$ with:

$$\mu: \mathcal{M} \otimes \mathcal{M} \to \mathcal{M} \tag{32}$$

$$\epsilon: 1 \to \mathcal{M} \tag{33}$$

such that the following diagrams commute:



Remark 19. Here the diagrams are required to commute up to equality. So if \mathcal{U} is the cartesian closed category of categories, a monoid in \mathcal{U} is a strict monoidal category.

Example 20. For \mathcal{U} the category of categories, a notion of parametricity is a (strict) monoidal category. We have many examples besides \Box from Definition 6. For example, all categories of cubes from [7] are monoidal. So we have notions of parametricity for categories corresponding to all kinds of cubes:

- We can have symmetries, diagonals or reflexivities. For reader familiar with cubical type theories, they correspond to structural rules on interval variables.
- We can have connections and inverses.

All notions of parametricity in Example 5 have reflexivities. Many more variants are possible, for example:

- We can consider the free monoidal category with an object \mathbb{I} and n maps to 1. This gives parametricity with n-ary relations rather than binary ones. For n = 1 we get unary parametricity, which is a form of iterated realisability.
- We can consider multiple generating objects in the monoidal category, for example bicubes are generated by two interval objects.

Remark 21. 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 send to a predicate A_{*} on some programs, and a proof of A is send to program p realising A (i.e. such that A_{*}(p) holds). For example A_* can be a predicate on untyped λ -terms, or Gödel numbers for recursive functions. This means that we can have a realiser for A without any proof of 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). This is sometimes summarised by saying that type theory is its own language of realisers. Since A_{*} is 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. shows formulas consistent by finding programs realising them.
- Unary parametricity emphasises the fact that terms are similar to programs, i.e. that they are continuous in some sense: they preserve the relevant predicates.

Remark 22. The monoidal category corresponding to unary parametricity is freely generated by an object \mathbb{I} and a map:

$$d: \mathbb{I} \to 1 \tag{34}$$

It is the (opposite of the) category of augmented semi-simplices. This means that augmented semi-simplices are unary semi-cubes!

There is a similar result for the category of augmented simplices, which is freely generated by a comonoid \mathbb{I} .

Remark 23. Recall that we used a non-standard orientation for the category of semi-cubes, so we should consider the opposites of the standard cube categories as notions of parametricity. This is not an issue because the opposite of a monoidal category is monoidal.

Now we can easily adapt our definition of parametric categories.

Remark 24. For C: U we have a monoid End_{C} in U with $C \multimap C$ as underlying object, composition as product and identity as unit.

Definition 25. An object C : U is called M-parametric if it is an M-module, *i.e.* if we are given a morphism of monoid from M to End_{C} .

A morphism from \mathcal{M} to $\operatorname{End}_{\mathcal{C}}$ can be reformulated as a map $\alpha : \mathcal{M} \otimes \mathcal{C} \to \mathcal{C}$ such that the following diagrams commute:





Remark 26. In the literature an \mathcal{M} -module is sometimes called an \mathcal{M} -action, depending on the context. We will sometimes say that \mathcal{M} acts on \mathcal{C} .

Maps between \mathcal{M} -modules respecting the \mathcal{M} -action are called *equivariant*. We give a precise definition:

Definition 27. An equivariant map between \mathcal{M} -modules (\mathcal{C}, α) and (\mathcal{D}, β) is a map:

$$f: \mathcal{C} \to \mathcal{D} \tag{35}$$

such that the following square commutes:



There is a category of modules and equivariant maps.

2.2 Freely and cofreely parametric objects

Let \mathcal{M} be a monoid in \mathcal{U} . Now we prove a theorem which will be used extensively in the rest of the paper. It can be summarised by saying that freely and cofreely \mathcal{M} -parametric objects always exist.

Theorem 1. The forgetful functor from \mathcal{M} -modules to \mathcal{U} 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 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 is true in any monoidal closed category. In order to check that we are indeed reasoning linearly, it is crucial that any bound variable in our λ -terms occurs precisely once. An alternative direct proof by diagram chasing in the symmetric monoidal closed category \mathcal{U} is certainly possible.

We proceed with the proof for the right adjoint. Given $\mathcal{C}:\mathcal{U}$, we define $R(\mathcal{C})$ as $\mathcal{M} \multimap \mathcal{C}$ with the \mathcal{M} -action α defined by:

$$\alpha : \mathcal{M} \multimap R(\mathcal{C}) \multimap R(\mathcal{C})$$
(36)

$$\alpha(i, u) = \lambda j. \, u(j \otimes i) \tag{37}$$

Which is indeed an action. Now for $f : \mathcal{C} \to \mathcal{D}$ we define:

$$R(f) : R(\mathcal{C}) \multimap R(\mathcal{D})$$
(38)

$$R(f,u) = f \circ u \tag{39}$$

We see that R(f) is equivariant, and that R is a functor from C to \mathcal{M} -modules.

Now we want to check that it is right adjoint to the forgetful functor. Assume (\mathcal{C}, α) an \mathcal{M} -action, and $\mathcal{D}: \mathcal{U}$. We define:

$$\psi : \operatorname{Hom}_{\mathcal{U}}(\mathcal{C}, \mathcal{D}) \to \operatorname{Hom}_{\mathcal{U}}(\mathcal{C}, \mathcal{M} \multimap \mathcal{D})$$

$$(40)$$

$$\psi(f) = \lambda c, i. f(\alpha(i, c)) \tag{41}$$

and check that $\psi(f)$ is equivariant. Next we define:

$$\phi : \operatorname{Hom}_{\mathcal{U}}(\mathcal{C}, \mathcal{M} \multimap \mathcal{D}) \to \operatorname{Hom}_{\mathcal{U}}(\mathcal{C}, \mathcal{D})$$
(42)

$$\phi(g) = \lambda c. \, g(c, 1) \tag{43}$$

We check that for all $f : \operatorname{Hom}_{\mathcal{U}}(\mathcal{C}, \mathcal{D})$ we have:

$$\phi(\psi(f)) = \lambda c. \, (\lambda c', i. f(\alpha(i, c')))(c, 1) \tag{44}$$

$$= \lambda c. f(\alpha(1, c)) \tag{45}$$

$$=f \tag{46}$$

and that for all equivariant $g : \operatorname{Hom}_{\mathcal{U}}(\mathcal{C}, \mathcal{M} \multimap \mathcal{D}))$, i.e. all g such that:

$$g(\alpha(i,c)) = \lambda j. \, g(c,j \otimes i) \tag{47}$$

we have:

$$\psi(\phi(g)) = \lambda c, i. (\lambda c'. g(c', 1))(\alpha(i, c))$$
(48)

$$= \lambda c, i. g(\alpha(i, c), 1) \tag{49}$$

$$=\lambda c, i. g(c, 1\otimes i) \tag{50}$$

$$=g \tag{51}$$

Now we just need naturality to conclude, so we check that for:

$$\mathcal{C}, \mathcal{C}', \mathcal{D}, \mathcal{D}' : \mathcal{U} \tag{52}$$

$$\alpha : \mathcal{M} \text{ acting on } \mathcal{C} \tag{53}$$

$$\alpha' : \mathcal{M} \text{ acting on } \mathcal{C}' \tag{54}$$

$$f : \operatorname{Hom}_{\mathcal{U}}(\mathcal{C}', \mathcal{C}) \text{ with } f \text{ equivariant}$$
 (55)

$$g: \operatorname{Hom}_{\mathcal{U}}(\mathcal{C}, \mathcal{D}) \tag{56}$$

$$h : \operatorname{Hom}_{\mathcal{U}}(\mathcal{D}, \mathcal{D}')$$
 (57)

we have that:

$$\psi(h \circ g \circ f) = \lambda c, i. (h \circ g)(f(\alpha'(i, c)))$$
(58)

$$= \lambda c, i. (h \circ g)(\alpha(i, f(c)))$$
(59)

$$= \lambda c. R(h)(\lambda i. g(\alpha(i, f(c))))$$
(60)

$$= R(h) \circ \psi(g) \circ f \tag{61}$$

We omit the similar proof for the left adjoint.
$$\Box$$

Example 28. For \mathcal{U} the category of categories and \Box the category of semi-cubes:

- The cofreely parametric category generated by C is the category of semicubical objects in C, i.e. the category of functor from □ to C.
- The freely parametric category generated by C is the category $\Box \times C$. The existence of this left adjoint is easy to prove, but this formula is still pleasantly explicit.

It works the same for all the previously mentioned variants of cubes, including bicubes and augmented simplices.

Remark 29. Recall that \mathcal{U} was simply assumed symmetric monoidal closed. Taking the category of abelian groups as \mathcal{U} , we get the following correspondence:

Model of type theory	Abelian groups
Notions of parametricity	Rings
Parametric models	Modules
Freely parametric models	Free modules
Cubical models	Cofree modules

Free (resp. cofree) modules are often called induced (resp. coinduced) modules.

3 Parametricity for left exact categories

In this section we apply the machinery from Section 2 to left exact categories (abbreviated as lex categories), i.e. categories with finite limits. So we will prove that the category of lex categories is symmetric monoidal closed, so that we can define notions of parametricity as monoids and parametric lex categories as modules. Then we will give examples of such monoids. It should be noted that we use a very strict version of lex categories:

- A lex category is given with a chosen limit for any finite diagram.
- Functors between lex categories have to preserve the chosen limits.
- Limits commute strictly, rather than up to natural isomorphisms.

These restrictions allow us to successfully use the 1-category of lex categories. For example it is immediate to see that we have adjoint functors freely adding and forgetting finite limits.

Remark 30. Already with categories in Example 20, we defined a notion of parametricity as a strict monoidal category, bringing some unnecessary strictness.

An alternative approach would be to use a symmetric monoidal closed 2category of models of type theory, with a notion of parametricity defined as a monoid where equations hold up to 2-isomorphisms.

Remark 31. Most notions of parametricity considered here are finitely presented, so that assuming them strict is painless.

3.1 The category of lex categories is symmetric monoidal closed

First we give our unusually strict definition of a lex category. We will require that some morphisms are identities, implicitly requiring that their sources and targets are equal.

Definition 32. A lex category is a category C with an operation \lim_{I} for any finite category I sending any $F : C^{I}$ to a limit cone.

Moreover we ask that for any finite categories I and J and diagram $H: \mathcal{C}^{I \times J}$, we have that the canonical isomorphisms:

$$\lim_{i:I} (\lim_{j:J} H(i,j)) \to \lim_{j:J} (\lim_{i:I} H(i,j))$$
(62)

are identities.

We define morphisms of lex categories as functor preserving the chosen limit cones. This gives a category Lex. It should be noted that this defines Lex as the category of algebras for an essentially algebraic extension of the theory of categories. In particular this means that the functor forgetting limits has a left adjoint freely adding them.

We define a symmetric monoidal closed structure on the category of lex categories. The arrow is immediate:

Definition 33. For C and D lex categories we define $C \multimap D$ as the category of lex functors from C to D with:

- Natural transformations as morphisms.
- Limits computed pointwise.

Remark 34. This definition would not be valid without the strict commutations of limits. Indeed for $(f_i)_{i:I}$ a finite diagram of lex functors, its pointwise limit being lex means that for any diagram $(x_j)_{j:J}$ we have that the canonical morphism:

$$\lim_{i:I} \lim_{j:J} f_i(x_j) \to \lim_{j:J} \lim_{i:I} f_i(x_j) \tag{63}$$

is an identity.

This phenomena is already present for groups: the pointwise product of two group morphisms is not necessarily 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 35. For C and D lex categories, we define $C \otimes D$ as the free lex category generated by a functor:

$$\underline{\quad} \otimes \underline{\quad} : \mathcal{C} \times \mathcal{D} \to \mathcal{C} \otimes \mathcal{D} \tag{64}$$

such that the canonical morphisms:

$$(\lim_{i:I} c_i) \otimes d \to \lim_{i:I} (c_i \otimes d) \tag{65}$$

$$c \otimes (\lim_{j:J} d_j) \to \lim_{j:J} (c \otimes d_j)$$
(66)

are identities.

This means that in order to define a a lex functor from $\mathcal{C} \otimes \mathcal{D}$ to \mathcal{E} , it is enough to define a functor $F : \mathcal{C} \times \mathcal{D} \to \mathcal{E}$ such that the morphisms:

$$F(\lim_{i:I} c_i, d) \to \lim_{i:I} F(c_i, d)$$
(67)

$$F(c, \lim_{j:J} d_j) \to \lim_{j:J} F(c, d_j)$$
(68)

are identities.

Remark 36. This definition implies that the canonical morphism:

$$\lim_{i:I} \lim_{j:J} (c_i \otimes d_j) \to \lim_{j:J} \lim_{i:I} (c_i \otimes d_j)$$
(69)

is an identity. This equality would be unnatural, although not incoherent, without the strict commutativity of limits.

Definition 37. We define 1 as the free lex category generated by an object.

So giving a lex functor from 1 to C is the same as giving an object in C.

Remark 38. 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 39. This is very similar to the tensor and arrow for abelian groups, with the following correspondence extending Remark 29.

Sets	Categories
Addition, zero	Finite limits
Abelian groups	$Lex\ categories$

We will make this formal in Remark 42.

We write U for the forgetful functor sending a lex category to its underlying category. We write F for its left adjoint freely adding limits. We prove an auxiliary lemma.

Lemma 40. For any category C and lex category D we have natural isomorphisms:

$$U(F(\mathcal{C}) \multimap \mathcal{D}) \cong U(\mathcal{D})^{\mathcal{C}}$$
(70)

Moreover limits in $F(\mathcal{C}) \multimap \mathcal{D}$ correspond to pointwise limits in $U(\mathcal{D})^{\mathcal{C}}$.

Proof. The isomorphism is immediate on objects. For any:

$$X : \operatorname{Hom}_{\operatorname{Lex}}(F(\mathcal{C}), \mathcal{D}) \tag{71}$$

we write \overline{X} the corresponding functor from \mathcal{C} to $U(\mathcal{D})$.

For \mathcal{D} a category, we denote by $\mathcal{D}^{\rightarrow}$ its arrow category with S (resp. T) the functor giving the source (resp. the target) of an arrow. Assume X and Y in $\operatorname{Hom}_{\operatorname{Lex}}(\mathcal{C}, \mathcal{D})$, then a natural transformation from X to Y is:

$$\{Z: \operatorname{Hom}_{\operatorname{Cat}}(\mathcal{C}, \mathcal{D}^{\to}) \mid S \circ Z = X, \ T \circ Z = G\}$$

$$(72)$$

But using the fact that X and Y are lex and naturality, we can show that any such Z is in fact lex. Then for any X and Y in $F(\mathcal{C}) \to \mathcal{D}$ we have:

$$\operatorname{Hom}_{F(\mathcal{C}) \to \mathcal{D}}(X, Y) \tag{73}$$

$$\cong \{ Z : \operatorname{Hom}_{\operatorname{Cat}}(F(\mathcal{C}), \mathcal{D}^{\to}) \mid S \circ Z = X, \ T \circ Z = Y \}$$
(74)

$$\cong \{ Z : \operatorname{Hom}_{\operatorname{Lex}}(F(\mathcal{C}), \mathcal{D}^{\to}) \mid S \circ Z = X, \ T \circ Z = Y \}$$

$$(75)$$

$$\cong \{ \overline{Z} : \operatorname{Hom}_{\operatorname{Lex}}(\mathcal{C}, U(\mathcal{D})^{\rightarrow}) \mid S \circ \overline{Z} = \overline{X}, \ T \circ \overline{Z} = \overline{Y} \}$$
(76)

$$\cong \operatorname{Hom}_{U(\mathcal{D})^{\mathcal{C}}}(\overline{X}, \overline{Y}) \tag{77}$$

This concludes the proof that:

$$U(F(\mathcal{C}) \multimap \mathcal{D}) \cong U(\mathcal{D})^{\mathcal{C}}$$
(78)

This isomorphism restricts functor and natural transformations defined on $F(\mathcal{C})$ to \mathcal{C} , and limits in $F(\mathcal{C}) \multimap \mathcal{D}$ are computed pointwise, so they correspond to pointwise limits in $U(\mathcal{D})^{\mathcal{C}}$.

Remark 41. In principle a natural transformation between lex functors should be assumed lex. This condition can be omitted (and always is) because it is always true, as used in the above proof.

Now we bring all these constructions together.

Theorem 2. This defines a symmetric monoidal closed structure on the category of lex categories.

Proof. There are many things to check:

• First we check that we have a natural isomorphism:

$$\operatorname{Hom}_{\operatorname{Lex}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{E}) \cong \operatorname{Hom}_{\operatorname{Lex}}(\mathcal{C}, \mathcal{D} \multimap \mathcal{E})$$
(79)

Indeed giving a morphism in:

$$\operatorname{Hom}_{\operatorname{Lex}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{E}) \tag{80}$$

is naturally equivalent to giving:

$$F: \operatorname{Hom}_{\operatorname{Cat}}(\mathcal{C} \times \mathcal{D}, \mathcal{E})$$
(81)

such that the canonical morphisms:

$$F(\lim_{i:I} c_i, d) \to \lim_{i:I} F(c_i, d)$$
(82)

$$F(c, \lim_{j:J} d_j) \to \lim_{j:J} F(c, d_j)$$
(83)

are identities, which is in turn naturally equivalent to giving:

$$\overline{F}: \operatorname{Hom}_{\operatorname{Cat}}(\mathcal{C}, \mathcal{E}^{\mathcal{D}})$$
(84)

such that the canonical morphisms:

$$\overline{F}(\lim_{i \in I} c_i)(d) \to \lim_{i \in I} \overline{F}(c_i)(d) \tag{85}$$

$$\overline{F}(c)(\lim_{j:J} d_j) \to \lim_{j:J} \overline{F}(c)(d_j)$$
(86)

are identities. The first conditions means that \overline{F} is a lex functor from \mathcal{C} to $\mathcal{E}^{\mathcal{D}}$, with limits in $\mathcal{E}^{\mathcal{D}}$ computed pointwise. The second condition means that the image of \overline{F} is included in $\mathcal{D} \multimap \mathcal{E}$, i.e. the full subcategory of $\mathcal{E}^{\mathcal{D}}$ consisting of lex functors. So together they precisely mean that:

$$\overline{F} : \operatorname{Hom}_{\operatorname{Lex}}(\mathcal{C}, \mathcal{E} \multimap \mathcal{D})$$
(87)

• Next we check that $_ \otimes _$ is symmetric. Indeed we can check that the the functor:

$$S : \mathcal{C} \times \mathcal{D} \to \mathcal{D} \otimes \mathcal{C} \tag{88}$$

$$S(c,d) = d \otimes c \tag{89}$$

commutes with limits in c and d, so that indeed it can be extended to:

$$S: \operatorname{Hom}_{\operatorname{Lex}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{D} \otimes \mathcal{C})$$
(90)

We can check that S is self-inverse.

• Similarly we can define a functor:

$$A : (\mathcal{C} \times \mathcal{D}) \times \mathcal{E} \to \mathcal{C} \otimes (\mathcal{D} \otimes \mathcal{E})$$
(91)

$$A((c,d),e) = c \otimes (d \otimes e) \tag{92}$$

which commutes with limits in each variable, so that it can be extended to:

$$A: \operatorname{Hom}_{\operatorname{Lex}}((\mathcal{C} \otimes \mathcal{D}) \otimes \mathcal{E}, \mathcal{C} \otimes (\mathcal{D} \otimes \mathcal{E}))$$
(93)

(94)

It is straightforward to define an inverse to A.

• Now we need to check that 1 is indeed a unit. But this is a consequence of the natural isomorphism:

$$(1 \multimap \mathcal{C}) \cong \mathcal{C} \tag{95}$$

in Lex, by Lemma 40 applied to $1 = F(\top)$.

We omit the necessary checking of the various coherence diagrams.

Remark 42. There exists a notion of commutative monad (see for example Section 6 in [5]). For T a commutative monad on a symmetric monoidal closed category C, the category of T-algebras is symmetric monoidal closed (assuming equalisers in C to build arrows and coequalisers in T-algebras to build tensors). We give two examples.

- The monad of abelian groups on sets is commutative.
- The monad of lex categories on categories is commutative, with our assumption:

$$\lim_{I} \lim_{J} = \lim_{J} \lim_{I} \tag{96}$$

Otherwise it would be commutative in a 2-categorical sense.

The monoidal closed structures on the categories of abelian groups and lex categories can both be built this way, cementing the analogy between abelian groups and lex categories from Remark 39.

3.2 Notions of parametricity for lex categories

Now we can use the symmetric monoidal closed structure from the previous section to define notions of parametricity as well as parametric models. We emphasise this:

Definition 43. A notion of parametricity for lex categories is a monoid in Lex.

Now we get the following proposition by unfolding this definition:

Proposition 44. Giving a notion of parametricity for lex categories is equivalent to giving a lex category \mathcal{M} with a (strictly) monoidal product $_ \otimes _$ such that the canonical morphisms:

$$(\lim_{i:I} m_i) \otimes n \to \lim_{i:I} (m_i \otimes n) \tag{97}$$

$$m \otimes (\lim_{j:J} n_j) \to \lim_{j:J} (m \otimes n_j)$$
(98)

are identities.

Such a category will be called a monoidal lex category. Now we give examples. First we prove that notions of parametricity for categories can be extended to lex categories.

Proposition 45. The functor F freely adding finite limits to a category is strongly monoidal.

Proof. We write U for the functor forgetting finite limits.

• We have a string of natural isomorphisms where C and D are categories and \mathcal{E} is a lex category:

$$\operatorname{Hom}_{\operatorname{Lex}}(F(\mathcal{C} \times \mathcal{D}), \mathcal{E}) \cong \operatorname{Hom}_{\operatorname{Cat}}(\mathcal{C} \times \mathcal{D}, U(\mathcal{E}))$$
(99)

$$\cong \operatorname{Hom}_{\operatorname{Cat}}(\mathcal{C}, U(\mathcal{E})^{\mathcal{D}})$$
(100)

$$\cong \operatorname{Hom}_{\operatorname{Cat}}(\mathcal{C}, U(F(\mathcal{D}) \multimap \mathcal{E}))$$
(101)

$$\cong \operatorname{Hom}_{\operatorname{Lex}}(F(\mathcal{C}), F(\mathcal{D}) \multimap \mathcal{E})$$
(102)

 $\cong \operatorname{Hom}_{\operatorname{Lex}}(F(\mathcal{C}) \otimes F(\mathcal{D}), \mathcal{E})$ (103)

where Equation 101 uses Lemma 40. We can conclude by Yoneda lemma.

• The lex category 1 is actually defined as $F(\top)$ with \top the terminal category.

We omit the necessary checking of the various coherence diagrams.

Corollary 46. For any notion of parametricity \mathcal{M} for categories, we have that $F(\mathcal{M})$ is a notion of parametricity for lex categories.

Proof. Strongly monoidal functors preserve monoids.

Now that we are considering lex categories, we can deal with truncated notions of parametricity, i.e. non-iterated ones. We fix n a natural number for this section. First we define the n-truncated semi-cubes:

Definition 47. Let \Box be the semi-cube category. Any object in \Box if of the from $\mathbb{I}^k = \mathbb{I} \otimes \cdots \otimes \mathbb{I}$ the k-th monoidal product of \mathbb{I} . We write \Box_n the full subcategory of \Box with objects $1, \mathbb{I}, \cdots, \mathbb{I}^n$.

Our goal is to show that \Box_n induces a notion of parametricity for lex categories. To do this we need to prove that the free lex category generated by \Box_n is a monoidal lex category.

Recall that the Day convolution extends $_ \otimes _$ on \square to a monoidal tensor on Set^{\square}. This tensor is closed on both sides, meaning that we have two bifunctors $_ \multimap _$ and $_ \multimap _$ with natural isomorphisms:

$$\operatorname{Hom}(X \otimes Y, Z) \cong \operatorname{Hom}(X, Y \multimap Z) \tag{104}$$

$$\operatorname{Hom}(X \otimes Y, Z) \cong \operatorname{Hom}(Y, Z \sim X) \tag{105}$$

for in $X, Y, Z : \operatorname{Set}^{\Box}$.

The inclusion of full subcategory $f:\Box_n\to\Box$ induces a post-composition functor:

$$f^* : \operatorname{Set}^{\square} \to \operatorname{Set}^{\square_n}$$
 (106)

with a full and faithful left (resp. right) adjoint $f_!$ (resp. f_*). An object $X: \text{Set}^{\square}$ is called coskeletal if:

$$X \cong f_*(f^*(X)) \tag{107}$$

and $f_*(f^*(X))$ is called the coskeleton of X.

Remark 48. Coskeletal cubical objects are usually called n-coskeletal, to emphasise the dependency on n. We omit n here because it is fixed for the whole section.

We want to give a helpful criteria for coskeletal object. First we define n-cells:

Definition 49. We define:

$$\delta \mathbb{I}^k : \mathrm{Set}^{\bigsqcup} \tag{108}$$

$$\delta i^k : \delta \mathbb{I}^k \to \mathbb{I}^k \tag{109}$$

by:

$$\delta i^0 : \bot \to 1 \tag{110}$$

and we define δi^{k+1} from δi^k using the pushout square:



where:

$$\alpha_X = (X \otimes d^0 \,|\, X \otimes d^1) \colon X \coprod X \to X \otimes \mathbb{I}$$
(111)

One can check that the map:

$$\delta i^k : \delta \mathbb{I}^k \to \mathbb{I}^k \tag{112}$$

is the inclusion of the border of a k-cell in the usual sense.

Definition 50. An morphism $u: A \to B$ is left orthogonal to an object X if the induced map:

$$u^* : \operatorname{Hom}(B, X) \to \operatorname{Hom}(A, X)$$
 (113)

is a bijection. We denote this by $u \perp X$.

This means that for any $f: A \to X$ there exists a unique dotted arrow making the triangle commutes:



We do not give a proof for the next lemma. It holds both for semi-cubes and cubes with referivities only (as claimed without proof in [15]), although we do not know to what extent it holds for other cubes.

Lemma 51. An element $X : \operatorname{Set}^{\Box}$ is coskeletal if and only δi^k for k > n is left orthogonal to X.

Lemma 52. Assume X coskeletal, then so are $\mathbb{I} \multimap X$ and $X \multimap \mathbb{I}$.

Proof. • We proceed with the proof for that $\mathbb{I} \multimap X$ is coskeletal when X is. We need to prove that:

$$\delta i^k \perp (\mathbb{I} \multimap X) \tag{114}$$

for k > n. This is equivalent to:

$$(\delta i^k \otimes \mathbb{I}) \perp X \tag{115}$$

We can decompose $\delta i^k \otimes \mathbb{I}$ as:

$$\delta \mathbb{I}^k \otimes \mathbb{I} \xrightarrow{\theta} \delta \mathbb{I}^{k+1} \xrightarrow{\delta i^{k+1}} \mathbb{I}^{k+1}$$

where θ is a pushout of $\delta i^k \coprod \delta i^k$. But since maps left orthogonal to X are stable by coproducts, pushouts and compositions, we can conclude.

• To give the proof for $X \sim \mathbb{I}$ we need to rework this section in mirror, with 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 product from $\operatorname{Set}^{\Box}$ to $\operatorname{Set}^{\Box_n}$

Lemma 53. The category $\operatorname{Set}^{\Box_n}$ inherits a monoidal structure from the Day convolution on $\operatorname{Set}^{\Box}$. This induced tensor on $\operatorname{Set}^{\Box_n}$ commutes with colimits in both variables.

Proof. We proceed in three steps.

• If Y is coskeletal, so are $\mathbb{I} \to Y$ and $Y \to \mathbb{I}$. This is Lemma 52, and this is the only part which rely on the kind of cube we use.

- If Y is coskeletal, so are $X \multimap Y$ and $Y \multimap X$. By iterating the previous point, the property is true when X is representable, and it is stable by colimits because coskeletal objects are stable by limits as f_* and f^* preserve limits.
- For any $X, Y : \operatorname{Set}^{\square}$ we have natural isomorphisms:

$$f^*(f_!f^*(X) \otimes Y) \cong f^*(X \otimes Y) \tag{116}$$

$$f^*(X \otimes f_! f^*(Y)) \cong f^*(X \otimes Y) \tag{117}$$

indeed we have for example:

$$\operatorname{Hom}(f^*(f_!f^*(X)\otimes Y), Z) \cong \operatorname{Hom}(f_!f^*(X)\otimes Y, f_*(Z))$$
(118)

$$\cong \operatorname{Hom}(f_! f^*(X), Y \multimap f_*(Z)) \quad (119)$$

$$\cong \operatorname{Hom}(X, f_*f^*(Y \multimap f_*(Z))) \quad (120)$$

$$\cong \operatorname{Hom}(X, Y \multimap f_*(Z)) \tag{121}$$

$$\cong \operatorname{Hom}(X \otimes Y, f_*(Z)) \tag{122}$$

$$\cong \operatorname{Hom}(f^*(X \otimes Y), Z) \tag{123}$$

where Equation 121 used the fact that $f_*(Z)$ is coskeletal, so that:

$$Y \multimap f_*(Z) \tag{124}$$

is coskeletal as well by the previous point.

• For $X, Y : \operatorname{Set}^{\Box_n}$, we can define:

$$X \otimes_n Y = f^*(f_!(X) \otimes f_!(Y)) \tag{125}$$

$$1_n = f^*(1) \tag{126}$$

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 $f_!$ commutes with colimits.

This proposition does depend on \Box when we check that if Y is coskeletal, so are $\mathbb{I} \multimap Y$ and $Y \multimap \mathbb{I}$. This is true for semi-cubes and cubes with reflexivities only, but we do not how far this can be extended.

Proposition 54. The free lex category generated by \Box_n is (non-strict) monoidal.

Proof. By duality, the previous lemma gives a monoidal structure $_ \otimes_n _$ on $(\operatorname{Set}^{\square_n})^{op}$, commuting with limits in both variables.

The free lex category generated by \Box_n is equivalent to the closure of representables in $(\text{Set}^{\Box_n})^{op}$ by finite limits. We want to restrict $_\otimes_n_$ to this closure. Since $_\otimes_n_$ commutes with finite colimits in both variables, it is sufficient to prove that $\mathbb{I}^k \otimes_n \mathbb{I}^{k'}$ is a finite colimit of representable in Set^{\Box_n} for $k, k' \leq n$ in order to conclude. We will prove the more general fact that $f^*(\mathbb{I}^l)$ is a finite limit of \mathbb{I}^k with $k \leq n$ for all l.

We know that for any object $X : \operatorname{Set}^{\Box_n}$ we have:

$$X \cong \operatorname{colim}_{i:\square_n^{op}, X(i)} \operatorname{Hom}_\square(i, _)$$
(127)

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 $X(i) = \text{Hom}_{\Box}(l, i)$ and \Box is locally finite), so that X is a finite limit of representables.

Here we used the fact that \Box was locally finite (i.e. has finite hom-sets) in a crucial way.

Remark 55. The monoidal category from the previous proposition is not strict, so that technically it is not a notion of parametricity. We expect that this can be worked around using a 2-category of models of type theory, or alternatively a strictification result for lex monoidal categories.

A \Box_n -parametric lex category has an endofunctor $__*$ such that for any X, the object $X_{*(n+1)}$ can be computed as a limit from X_{*k} with $k \leq n$. A cofreely \Box_n -parametric lex category is simply a category of *n*-cubical objects in some lex category.

Remark 56. In the usual type-theoretic point of view on parametricity, the condition on $__*$ can be reformulated simply as $X_{*(n+1)} = \top$, as the limit involved is taken care of by the complicated context in which $X_{*(n+1)}$ is defined.

We conjecture that this n-truncated parametricity for lex categories actually works for all kinds of cubes.

4 Parametricity for clans

The notion of clans was designed by Joyal to help bridge the gap between type theory and homotopy theory [13]. It is based on the idea that types can be modeled by fibrations as follows:

- A fibration $f: A \to \Gamma$ is a map with fibers varying continuously in Γ .
- A dependent type over Γ is family of types varying continuously in Γ .

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

Axiomatising the notion of fibration has been done in many different ways in the past, notably in model categories [18] and categories of fibrant objects [6]. Fibrations in clans retain very little features from these classical homotopical axiomatisations.

There is no major new idea compared to the last section, but everything is technically more involved. First we will give our unusually strict definition of clans and show that the category of clans is symmetric monoidal closed. Then we will give examples of notions of parametricity for clans and examine the corresponding cubical models.

4.1 Clans with strictly commuting limits

Here we give our definition of clans. We define the usual clans and call them weak clans, so that we can define clans as weak clans with strictly commuting limits. First we give an auxiliary definition.

Definition 57. A class of maps \mathcal{F} is called closed by pullbacks if for any:

$$f: c \to b$$
 (128)

$$g: a \to b \tag{129}$$

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

$$\begin{array}{c|c} a \times c \xrightarrow{\pi_c} c \\ & \downarrow \\ \pi_a \\ a \xrightarrow{g} b \end{array} \downarrow f$$

where π_a is in \mathcal{F} . For any commutative square:

$$\begin{array}{c} d \xrightarrow{q} c \\ \downarrow \\ p \\ a \xrightarrow{g} b \end{array}$$

we write:

$$\langle p,q \rangle : d \to a \underset{b}{\times} c$$
 (130)

the unique map such that:

$$\pi_a \circ \langle p, q \rangle = p \tag{131}$$

$$\pi_c \circ \langle p, q \rangle = q \tag{132}$$

Recall that clans attempt to capture the intuition of a class of maps with continuously varying fibers.

Definition 58. A weak clan is a category with a terminal object \top , together with a class of maps called fibrations such that:

- Fibrations are closed by isomorphisms, compositions and pullbacks.
- Maps to \top are fibrations.

We write:

$$f:a \twoheadrightarrow b$$
 (133)

to say that f is a fibration.

Remark 59. The map id_a is isomorphic to the pullback of id_{\top} along the unique map from a to \top , so that it is a fibration.

Remark 60. Cartesian products are defined in any weak clan as pullbacks:



So cartesian projections are fibrations. They correspond to non-dependent types.

Remark 61. We have a correspondence between weak clans and type theory, where a type over a context Γ is a fibration with target Γ . We explain assumptions on fibrations from this point of view:

Fibrations	Types
Pullbacks	Substitutions
Identities	Т
Compositions	Σ
Maps to \top	Democracy
Projections	$Non-dependent\ types$

Now we want to define strictly commuting limits for clans. We recall some vocabulary:

Definition 62. A square:

$$\begin{array}{c} d \longrightarrow c \\ \downarrow \\ \downarrow \\ a \longrightarrow b \end{array}$$

is called Reedy fibrant if the induced map:

$$d \to a \underset{b}{\times} c \tag{134}$$

is a fibration.

Now we can define clans.

Definition 63. A clan is a weak clan with strictly commuting limits. This means that:

• The canonical morphism:

$$\top \to \top \underset{\top}{\times} \top \tag{135}$$

is the identity.

• For any diagram:



where the bottom right square is Reedy fibrant, the canonical morphism between its limit computed row-by-row or column-by-column is the identity.

Proof. To check that this is meaningful, we have to check that for any diagram:

$$\begin{array}{c} a \longrightarrow b \longleftarrow c \\ \downarrow & \downarrow & \downarrow \\ a' \longrightarrow b' \longleftarrow c' \end{array}$$

with the left square Reedy fibrant, we have that the induced map:

$$a \underset{b}{\times} c \to a' \underset{b'}{\times} c' \tag{136}$$

is a fibration. Indeed the following diagram shows that it is isomorphic to the composite of pullbacks of fibrations.



 \Box

We will see how strictly commutative limits ensure the existence of a linear arrow for clans. As for lex categories this restriction is necessary because we consider morphisms of clans commuting with limits on the nose. We could presumably use the 2-category of weak clans, without further assumptions on limits. Now we give an auxiliary lemma on free clans.

Lemma 64. Fibrations in the free clan are maps isomorphic to projections.

Proof. It is straightforward to see that any clan has cartesian products, and that maps isomorphic to cartesian projections are always fibrations. Now we prove that the class of maps isomorphic to a projection is a valid class of fibrations:

- They are stable by isomorphisms by definition.
- They are stable by compositions as for any objects a, b, c we have:

• They are stable by pullbacks because for any objects a, b, c and morphism $\sigma : c \to a$, the square:

$$\begin{array}{c} c \times b \xrightarrow{\sigma \times b} a \times b \\ \pi_c \downarrow & \downarrow \pi_a \\ c \xrightarrow{\sigma} a \end{array}$$

is a pullback square.

• Maps to \top are isomorphic to projections:

$$a \xrightarrow{\cong} \top \times a \xrightarrow{\pi_{\top}} \top$$

So fibrations are precisely maps isomorphic to projections.

4.2 Monoidal structure on clans

Now we define the tensor product, the linear arrow and the unit of clans.

Definition 65. Let C and D be clans. We define a clan $C \multimap D$ where:

- Objects are morphisms of clans.
- Morphisms are natural transformations between the underlying functors.
- Limits are computed pointwise.
- A natural transformation $\alpha: F \to G$ is a fibration if for any fibration $c \twoheadrightarrow c'$ in \mathcal{C} we have an induced fibration in \mathcal{D} :

$$F(c) \twoheadrightarrow F(c') \underset{G(c')}{\times} G(c)$$
(137)

Proof. We need to check that this is indeed a clan. As a preliminary result, we prove that fibrations are pointwise fibrations. Indeed for:

$$\alpha: F \twoheadrightarrow G \tag{138}$$

a fibration and c in C, from the fact that $c \twoheadrightarrow \top$ we have a fibration:

$$H(c) \twoheadrightarrow H(\top) \underset{G(\top)}{\times} G(c)$$
(139)

but this map is isomorphic to the map $\alpha(c): H(c) \to G(c)$. Now we check that limits are well defined:

• The constant functor F_{\top} with value \top is a morphism of clan, giving a terminal object in $\mathcal{C} \multimap \mathcal{D}$. The main thing to check is that the induced map:

$$F_{\top}(a \underset{b}{\times} c) \to F_{\top}(a) \underset{F_{\top}(b)}{\times} F_{\top}(c)$$
(140)

is an identity but this map is equal to the unique map in:

$$\top \to \top \underset{\top}{\times} \top \tag{141}$$

which is an identity.

• Now we check that pullbacks can be defined pointwise in $\mathcal{C} \multimap \mathcal{D}$. So we assume given a diagram in $\mathcal{C} \multimap \mathcal{D}$ as follows:

$$F \longrightarrow G \iff H$$

And we define $F \underset{G}{\times} H$ by:

$$(F \underset{G}{\times} H)(c) = F(c) \underset{G(c)}{\times} H(c)$$
(142)

We check that it commutes with limits. We need to check that the morphism:

$$(F \underset{G}{\times} H)(\top) \to \top \tag{143}$$

is an identity so that $F \underset{G}{\times} H$ preserves the terminal object. But this map is the inverse to:

$$\top \to \top \underset{\top}{\times} \underset{\top}{\top}$$
(144)

which is assumed to be an identity.

To check that $F \underset{G}{\times} H$ commutes with pullbacks, we consider a diagram:

$$a \longrightarrow b \iff c$$

in \mathcal{C} . We need to prove that the canonical morphism from the limit of the diagram:



computed row-by-row and column-by-column is an identity. This is a special case of strict commutation of limits because the bottom left square is Reedy fibrant by the definition of the fibration $H \rightarrow G$.

Next we check that the class of fibrations is suitably closed.

- Stability by isomorphisms is straightforward.
- Compositions of fibrations are fibrations. Indeed for a diagram:

$$F \longrightarrow G \longrightarrow H$$

we need to prove that for any fibration $c \twoheadrightarrow c'$ in C the map:

$$F(c) \to F(c') \underset{H(c')}{\times} H(c)$$
(145)

is a fibration in \mathcal{D} . But this map is a composite of fibrations as follows:

• Pullbacks of fibrations are fibrations. We have already checked that they exist and are computed pointwise. Now we need to check that given a diagram:

$$F \longrightarrow G \longleftarrow H$$

in $\mathcal{C} \multimap \mathcal{D}$ the map:

$$F \underset{G}{\times} H \to F \tag{146}$$

is a fibration, so that we need to prove that for any fibration $c \twoheadrightarrow c'$ in $\mathcal{C},$ the induced map:

$$(F \underset{G}{\times} H)(c) \to F(c) \underset{F(c')}{\times} (F \underset{G}{\times} H)(c')$$
(147)

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

$$F(c) \underset{G(c)}{\times} H(c) \to F(c) \underset{G(c')}{\times} H(c')$$
(148)

which is isomorphic to a pullback of fibration as follows:

• Maps to \top are fibrations, indeed the unique map:

$$F \to \top$$
 (149)

is a fibration if for any fibration:

$$c \twoheadrightarrow c'$$
 (150)

in \mathcal{C} the map:

$$F(c) \to F(c') \underset{\top}{\times} \top$$
 (151)

is a fibration in \mathcal{D} . But this map is isomorphic to the map:

$$F(c) \to F(c') \tag{152}$$

which is a fibration because F preserves fibrations.

Finally we can check that limits are strictly commuting because they are computed pointwise. $\hfill \Box$

Now we define the tensor product of two clans.

Definition 66. Given class C and D, we define $C \otimes D$ as the free clan generated by:

• A functor:

$$\underline{} \otimes \underline{} : \mathcal{C} \times \mathcal{D} \to \mathcal{C} \otimes \mathcal{D} \tag{153}$$

• The fact that for any two fibrations:

$$c \twoheadrightarrow c'$$
 (154)

$$d \twoheadrightarrow d'$$
 (155)

we have a fibration:

$$c \otimes d \twoheadrightarrow (c' \otimes d) \underset{c' \otimes d'}{\times} (c \otimes d')$$
(156)

• For any object c : C and d : D the canonical morphisms:

$$c \otimes \top \to \top \tag{157}$$

$$\top \otimes d \to \top \tag{158}$$

 $are \ identities.$

• Given a span:

$$c_1 \longrightarrow c_2 \longleftarrow c_3$$

in C and a span:

$$d_1 \longrightarrow d_2 \lll d_3$$

in \mathcal{D} with $c: \mathcal{C}$ and $d: \mathcal{D}$, the canonical morphisms:

$$(c_1 \underset{c_2}{\times} c_3) \otimes d \to (c_1 \otimes d) \underset{c_2 \otimes d}{\times} (c_3 \otimes d)$$
(159)

$$c \otimes (d_1 \underset{d_2}{\times} d_3) \to (c \otimes d_1) \underset{c \otimes d_2}{\times} (c \otimes d_3)$$
(160)

are identities.

Proof. To check that the last axiom makes sense, we need to show that for any fibration $c \rightarrow c'$ in C and d and object in \mathcal{D} we have a fibration:

$$c \otimes d \twoheadrightarrow c' \otimes d \tag{161}$$

and the same with the role of ${\mathcal C}$ and ${\mathcal D}$ reversed. This is true because:

• By the second assumption in $\mathcal{C} \otimes \mathcal{D}$ applied to $c \twoheadrightarrow c'$ and $d \twoheadrightarrow \top$ we have:

$$c \otimes d \twoheadrightarrow (c' \otimes d) \underset{c' \otimes \top}{\times} (c \otimes \top)$$
(162)

• By the third assumption, this fibration is isomorphic to:

$$c \otimes d \twoheadrightarrow c' \otimes d \tag{163}$$

Remark 67. In homotopy theory the usual condition on a tensor is dual, i.e. it is a condition of stability for cofibrations, involving a pushout rather than a pullback. This is not surprising because we use the opposite from the standard categories of cubes.

Finally we define the unit.

Definition 68. We define 1 as the free clan generated by an object.

Remark 69. The clan 1 is the opposite to the category of finite sets, with monomorphisms as fibrations.

And now we are ready to prove that this defines a monoidal structure. We will proceed exactly as for lex categories, with a few more properties to check in order to take fibrations into account. We denote by F the left adjoint to the forgetful functor U from clans to categories.

Lemma 70. For any category C and clan D we have a natural isomorphism:

$$U(F(\mathcal{C}) \multimap \mathcal{D}) \cong U(\mathcal{D})^{\mathcal{C}}$$
(164)

Moreover:

- Fibrations in F(C) → D correspond to natural transformations in U(D)^C which are pointwise fibrations.
- Limits in $F(\mathcal{C}) \longrightarrow \mathcal{D}$ correspond to pointwise limits in $U(\mathcal{D})^{\mathcal{C}}$.

Proof. The natural isomorphism is very similar to the proof for lex categories in Lemma 40, using the clan structure on $\mathcal{D}^{\rightarrow}$ with the fibrations defined pointwise. The new result we need to prove is that for any two morphisms of clans $X, Y : F(\mathcal{C}) \to \mathcal{D}$ with:

$$Z : \operatorname{Hom}_{\operatorname{Cat}}(F(\mathcal{C}), \mathcal{D}^{\to})$$
(165)

$$S \circ Z = X \tag{166}$$

$$T \circ Z = Y \tag{167}$$

we have that Z sends fibrations to pointwise fibrations in $\mathcal{D}^{\rightarrow}$. This means that for any fibration $i: c \twoheadrightarrow c'$ in $F(\mathcal{C})$ we have that the vertical maps in:

$$\begin{array}{c|c} X(c) \xrightarrow{Z(c)} Y(c) \\ X(i) & & & \downarrow Y(i) \\ X(c') \xrightarrow{Z(c')} Y(c') \end{array}$$

are fibrations. This is true because X and Y preserves fibrations.

Now we want to check that fibrations in $F(\mathcal{C}) \multimap \mathcal{D}$ corresponds to pointwise fibrations in $U(\mathcal{D})^{\mathcal{C}}$. Assume given:

$$X, Y: F(\mathcal{C}) \multimap \mathcal{D} \tag{168}$$

$$\alpha : \operatorname{Hom}_{F(\mathcal{C}) \multimap \mathcal{D}}(X, Y) \tag{169}$$

with:

$$\overline{X}, \overline{Y}: U(\mathcal{D})^{\mathcal{C}}$$
(170)

$$\overline{\alpha} : \operatorname{Hom}_{U(\mathcal{D})^{\mathcal{C}}}(\overline{X}, \overline{Y})$$
(171)

the corresponding elements in $U(\mathcal{D})^{\mathcal{C}}$. The following are equivalent:

- The morphism α is a fibration.
- For all $c \twoheadrightarrow c'$ in $F(\mathcal{C})$, we have an induced fibration:

$$X(c) \twoheadrightarrow X(c') \underset{Y(c')}{\times} Y(c)$$
(172)

• For all c and c' in $F(\mathcal{C})$, we have an induced fibration:

$$X(c) \times X(c') \twoheadrightarrow X(c) \times Y(c') \tag{173}$$

(because any fibration in $F(\mathcal{C})$ is isomorphic to a projection $c \times c' \to c$ by Lemma 64 and we have:

$$\begin{array}{c|c} X(c \times c') & \longrightarrow & X(c) \underset{Y(c)}{\times} Y(c \times c') \\ & \cong & & \downarrow \cong \\ X(c) \times X(c') & \longrightarrow & X(c) \times Y(c') \end{array}$$

so the top arrow is a fibration if and only if the bottom one is).

• For all c' in $F(\mathcal{C})$ we have an induced fibration:

$$X(c') \twoheadrightarrow Y(c') \tag{174}$$

(because $X(c') \twoheadrightarrow Y(c')$ implies that for any $d : \mathcal{D}$ we have a fibration:

$$d \times X(c') \twoheadrightarrow d \times Y(c') \tag{175}$$

and we consider d = X(c).

• For all c in C we have an induced fibration:

$$\overline{X}(c) \twoheadrightarrow \overline{Y}(c) \tag{176}$$

(because X and Y are the extensions of \overline{X} and \overline{Y} commuting with cartesian products, so the condition implies that $X(c) \twoheadrightarrow Y(c)$ for all $c = c_1 \times \cdots \times c_n$ and any object in $F(\mathcal{C})$ is of this form).

• The morphism $\overline{\alpha}$ is a pointwise fibration.

The fact that limits are computed pointwise is straightforward. $\hfill \Box$

Theorem 3. This defines a symmetric monoidal closed structure on the category of clans.

Proof. The proof is an extension of the proof of Theorem 2 from lex categories to clans, where we have to check at every step that it can be adapted to fibrations.

• Concerning the isomorphisms:

$$\operatorname{Hom}_{\operatorname{Clan}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{E}) \cong \operatorname{Hom}_{\operatorname{Clan}}(\mathcal{C}, \mathcal{D} \multimap \mathcal{E})$$
(177)

We know from the lex case that giving a limit preserving functor:

$$F: \operatorname{Hom}_{\operatorname{Cat}}(\mathcal{C} \otimes \mathcal{D}, \mathcal{E}) \tag{178}$$

is equivalent to giving a limit preserving:

$$\overline{F}: \operatorname{Hom}_{\operatorname{Cat}}(\mathcal{C}, \mathcal{D} \multimap \mathcal{E})$$
(179)

so it is enough to check that F preserves fibrations if and only if so does \overline{F} .

We have that F is preserves fibrations if for all fibration $c \twoheadrightarrow c'$ in Cand $d \twoheadrightarrow d'$ in \mathcal{D} we have a fibration:

$$F(c \otimes d) \twoheadrightarrow F((c' \otimes d) \underset{c' \otimes d'}{\times} (c \otimes d'))$$
(180)

$$= F(c' \otimes d) \underset{F(c' \otimes d')}{\times} F(c \otimes d')$$
(181)

in \mathcal{E} .

On the other hand \overline{F} preserves fibrations if and only if for any fibration $c \twoheadrightarrow c'$ in C we have a fibration:

$$\overline{F}(c) \twoheadrightarrow \overline{F}(c') \tag{182}$$

in $\mathcal{D} \multimap \mathcal{E}$, which means precisely that for all fibration $d \twoheadrightarrow d'$ we have a fibration:

$$\overline{F}(c,d) \twoheadrightarrow \overline{F}(c,d') \underset{\overline{F}(c',d')}{\times} \overline{F}(c',d)$$
(183)

so we see that both conditions are equivalent.

• Next we check symmetry. To extend the result for lex categories, we need to check that:

$$S: \mathcal{C} \otimes \mathcal{D} \to \mathcal{D} \otimes \mathcal{C} \tag{184}$$

preserves fibrations, meaning that for fibrations $c \twoheadrightarrow c'$ in C and $d \twoheadrightarrow d'$ in \mathcal{D} we have an induced fibration:

$$S(c \otimes d) \twoheadrightarrow S((c' \otimes d) \underset{c' \otimes d'}{\times} (c \otimes d'))$$
(185)

but this map is equal to:

$$d \otimes c \twoheadrightarrow (d \otimes c') \underset{d' \otimes c'}{\times} (d' \otimes c)$$
(186)

$$\cong (d' \otimes c) \underset{d' \otimes c'}{\times} (d \otimes c') \tag{187}$$

which is a fibration in $\mathcal{D}\otimes\mathcal{C}$.

• Then we check associativity. To extend the result for lex categories, we need to check that:

$$A: (\mathcal{C} \otimes \mathcal{D}) \otimes \mathcal{E} \to \mathcal{C} \otimes (\mathcal{D} \otimes \mathcal{E})$$
(188)

preserves fibrations, i.e. that for any fibrations $c \twoheadrightarrow c'$ in C, with $d \twoheadrightarrow d'$ in D and $e \twoheadrightarrow e'$ in \mathcal{E} , we have an induced fibration:

$$A((c \otimes d) \otimes e) \twoheadrightarrow A\big(((c \otimes d)' \otimes e)'\big)$$
(189)

where we use the informal notation:

$$(x \otimes y)' = x' \otimes y \underset{x' \otimes y'}{\times} x \otimes y'$$
(190)

but the maps:

$$(c \otimes d) \otimes e \twoheadrightarrow ((c \otimes d)' \otimes e)'$$
(191)

$$c \otimes (d \otimes e) \twoheadrightarrow \left(c \otimes (d \otimes e)' \right)' \tag{192}$$

are isomorphic as their right-hand sides are isomorphic to the limits of the diagrams:

$$(c' \otimes d) \otimes e \qquad (c \otimes d') \otimes e \qquad (c \otimes d) \otimes e'$$
$$(c \otimes d') \otimes e' \qquad (c' \otimes d) \otimes e' \qquad (c' \otimes d') \otimes e$$

and:

$$\begin{array}{ccc} c'\otimes (d\otimes e) & c\otimes (d'\otimes e) & c\otimes (d\otimes e') \\ & & \\ c\otimes (d'\otimes e') & c'\otimes (d\otimes e') & c'\otimes (d'\otimes e) \end{array}$$

• Finally we check the unit isomorphisms using the fact that:

$$(1 \multimap \mathcal{C}) \cong \mathcal{C} \tag{193}$$

by Lemma 70 applied to $1 = F(\top)$ with \top the terminal category.

4.3 Notions of parametricity for clans

Now we can apply our machinery to define a notion of parametricity for clans as a monoidal clan. Explicitly we have:

Corollary 71. A notion of parametricity for clans is a clan C with a strictly monoidal product $_ \otimes _$ such that:

• For any two fibrations:

$$c \twoheadrightarrow c'$$
 (194)

$$d \twoheadrightarrow d'$$
 (195)

we have an induced fibration:

$$c \otimes d \twoheadrightarrow (c' \otimes d) \underset{c' \otimes d'}{\times} (c \otimes d')$$
(196)

• For any object c : C the canonical morphisms:

$$c \otimes \top \to \top \tag{197}$$

are identities.

• Given a span:

$$c_1 \longrightarrow c_2 \twoheadleftarrow c_3$$

in C with d:C, the canonical morphisms:

$$(c_1 \underset{c_2}{\times} c_3) \otimes d \to (c_1 \otimes d) \underset{c_2 \otimes d}{\times} (c_3 \otimes d)$$
(199)

$$d \otimes (c_1 \underset{c_2}{\times} c_3) \to (d \otimes c_1) \underset{d \otimes c_2}{\times} (d \otimes c_3)$$
(200)

are identities.

We now give some examples. First we extend notions of parametricity for categories to clans.

Proposition 72. The functor F building free clans is monoidal.

Proof. This is exactly the same as Lemma 45 for lex categories, using Lemma 70 instead of Lemma 40. $\hfill \Box$

Corollary 73. If C is a notion of parametricity for categories, then F(C) is a notion of parametricity for clans. The cubical model:

$$F(\mathcal{C}) \multimap \mathcal{D} \tag{201}$$

is isomorphic to the clan where:

- Objects are functors from C to D.
- Morphisms are natural transformations.
- Limits and fibrations are defined pointwise.

Proof. This is a reformulation of Lemma 70.

Now we introduce a genuinely new notion of parametricity.

Definition 74. Let \Box be a monoidal category with:

$$\mathbb{I}:\Box\tag{202}$$

$$d^0, d^1: \mathbb{I} \to 1 \tag{203}$$

such that any object in \Box is of the form \mathbb{I}^n for some $n \geq 0$.

From this data we define a notion of parametricity for clans \Box_c as the free monoidal clan generated by:

- A monoidal functor from \Box to \Box_c
- The fact that we have a fibration:

$$\langle d^0, d^1 \rangle : \mathbb{I} \twoheadrightarrow 1 \times 1$$
 (204)

Definition 75. A notion of parametricity for clans is called cubical if it comes from the previous definition.

Example 76. The main example is the free monoidal clan generated by:

- An object \mathbb{I} .
- A fibration $i : \mathbb{I} \rightarrow 1 \times 1$.

It is called the clan of semi-cubes. This can be adapted to any variant of cubes.

Remark 77. The clan of semi-cubes corresponds to the usual parametricity for type theory, reformulated in the language of clans. Indeed the map:

$$i: \mathbb{I} \to 1 \times 1$$
 (205)

is a fibration so that in any parametric clan C we have:

• For any object $\Gamma : C$ we have a fibration:

$$a_* \twoheadrightarrow a \times a$$
 (206)

Type-theoretically this means that any context $\Gamma \vdash$ comes with a relation:

$$\Gamma, \Gamma \vdash \Gamma_*$$
 (207)

• For a fibration $a \rightarrow b$ in C, we have a fibration:

$$a_* \twoheadrightarrow b_* \times_{b \times b} (a \times a) \tag{208}$$

Type-theoretically this means that for $\Gamma \vdash A$ we have:

$$\Gamma, \Gamma, \Gamma_*, A, A \vdash A_* \tag{209}$$

as is customary with parametricity.

We give some notations:

Definition 78. Given a clan C and a fibration $i: a \rightarrow b$ in C, we define:

$$a^{n}: Ob_{\mathcal{C}}$$
(210)

$$\delta a^{n} : Ob_{\mathcal{C}}$$
(210)
$$\delta i^{n} : a^{n} \twoheadrightarrow \delta a^{n}$$
(212)

$$\partial i^n : a^n \twoheadrightarrow \partial a^n \tag{212}$$

by:

$$a^0 = 1 \tag{213}$$

$$a^{n+1} = a^n \otimes A \tag{214}$$

and:

$$\delta a^0 = \top \tag{215}$$

$$\delta a^{n+1} = (\delta a^n) \otimes a \underset{(\delta a^n) \otimes b}{\times} a^n \otimes b \tag{216}$$

with:

$$\delta i^0 = \epsilon_1 : 1 \to \top \tag{217}$$

$$\delta i^{n+1} = \delta i^n \odot i \tag{218}$$

where for:

$$f: x \twoheadrightarrow x' \tag{219}$$

$$g: y \twoheadrightarrow y' \tag{220}$$

we write:

$$f \odot g : x \otimes y \twoheadrightarrow (x \otimes y') \underset{x' \otimes y'}{\times} (x' \otimes y)$$
(221)

Now we can state the key lemma about cubical notions of parametricity.

Lemma 79. Assume given a cubical notions of parametricity \Box_c built from \Box , I, d^0 , d^1 . The clan underlying \Box_c is the free clan generated by:

• A functor from the category \Box to \Box_c .

• The fact that δi^n is a fibration for all $n \ge 0$, where $i = \langle d^0, d^1 \rangle$.

Proof. We will show that both considered clans are isomorphic to an intermediate free clan \Box' generated by:

- A morphism of clan from the free monoidal clan generated by \Box to \Box' .
- The fact that δi^n is a fibration in \Box' for all $n \ge 0$.

First we check that asking a morphism of clan from the free monoidal clan generated by the monoidal category \Box is the same as asking for a functor from \Box . Indeed the following square commutes up to natural isomorphism:

$$\begin{array}{ccc} \operatorname{Cat} & \overline{U} & \operatorname{MonCat} \\ F & & & & \\ F & & & \\ \operatorname{Clan} & \underbrace{U} & \operatorname{MonClan} \end{array}$$

because F is strongly monoidal by Lemma 72. So we have natural isomorphisms:

$$\operatorname{Hom}_{\operatorname{Clan}}(U\overline{F}(\Box), \mathcal{D}) \cong \operatorname{Hom}_{\operatorname{Clan}}(F\overline{U}(\Box), \mathcal{D})$$
(222)

$$\cong \operatorname{Hom}_{\operatorname{Cat}}(\overline{U}(\Box), \mathcal{D})$$
 (223)

This means that the intermediate clan \Box' is equivalent to the free clan generated by a functor from \Box and δi^n a fibration for all $n \ge 0$.

Next we prove that in the free monoidal clan generated by the monoidal category \Box the following conditions are equivalent:

- δi^n is a fibration for all $n \ge 0$.
- *i* is a fibration and fibrations are stable by $_\odot_$.

This will implies that \Box_c is isomorphic to the intermediate clan \Box' .

The fact that the second condition implies the first is clear, as δi^0 is always a fibration and:

$$\delta i^{n+1} = \delta i^n \odot i \tag{224}$$

For the converse we assume that δi^n is a fibration for all $n \ge 0$, and we want to check that fibrations are closed by $_\odot_$. We prove inductively on the fibration ψ that for any fibration $f: x \twoheadrightarrow y$, we have that $f \odot \psi$ is a fibration:

• For $g: a \to b$ and $g': b \to c$ with $\psi = g' \circ g$, we have a commutative square:

$$\begin{array}{c} x \otimes a \xrightarrow{f \odot (g' \circ g)} & (y \otimes a) \underset{y \otimes c}{\times} (x \otimes c) \\ f \odot g \downarrow & \downarrow \cong \\ (y \otimes a) \underset{y \otimes b}{\times} (x \otimes b) \xrightarrow{(y \otimes a) \underset{y \otimes b}{\times} (f \odot g')} & (y \otimes a) \underset{y \otimes b}{\times} (y \otimes b) \underset{y \otimes c}{\times} (x \otimes c) \end{array}$$

so that if $f \odot g$ and $f \odot g'$ are fibrations, then so is $f \odot (g' \circ g)$.

• For $h: a \to b$ and $g: c \to b$ with $\psi = \pi_a$ we have a pullback square:



and a commutative square:

$$\begin{array}{c} x \otimes (a \underset{b}{\times} c) \xrightarrow{\qquad f \odot \pi_a \qquad} (x \otimes a) \underset{y \otimes a}{\times} (y \otimes (a \underset{b}{\times} c)) \\ \cong \downarrow \qquad \qquad \downarrow \qquad \qquad \downarrow \\ (x \otimes a) \underset{x \otimes b}{\times} (x \otimes c) \qquad (x \otimes a) \underset{y \otimes a}{\times} (y \otimes a) \underset{y \otimes b}{\times} y \otimes c \\ \xrightarrow{(x \otimes a) \underset{x \otimes b}{\times} (f \odot g)} \downarrow \qquad \qquad \downarrow \\ (x \otimes a) \underset{x \otimes b}{\times} (x \otimes b) \underset{y \otimes b}{\times} (y \otimes c) \xrightarrow{\qquad = \ } (x \otimes a) \underset{y \otimes b}{\times} (y \otimes c) \end{array}$$

so that if $f \odot g$ is a fibration, then so is $f \odot \pi_a$.

• For a an object we have for $\epsilon_a : a \to \top$ a commutative square:

$$\begin{array}{c|c} x \otimes a & \xrightarrow{f \odot \epsilon_a} & (y \otimes a) \underset{y \otimes \top}{\times} (x \otimes \top) \\ f \otimes a & \downarrow & \downarrow \cong \\ y \otimes a & \xrightarrow{\cong} & (y \otimes a) \underset{\top}{\times} \top \end{array}$$

and then we reason inductively on a, using the following isomorphisms:

$$f \otimes 1 \cong f \tag{225}$$

$$f \otimes (a \otimes b) \cong (f \otimes a) \otimes b \tag{226}$$

$$f \otimes \top \cong \mathrm{id}_{\top} \tag{227}$$

$$f \otimes \top \cong \operatorname{id}_{\top}$$

$$f \otimes (a \underset{b}{\times} c) \cong (f \otimes a) \underset{f \otimes b}{\times} (f \otimes c)$$
(228)

Finally we have a commuting triangle:



so that if $f \odot i$ is a fibration then so is $f \otimes \mathbb{I}$, and in turn so is $f \odot \epsilon_a$ for all object a.

• We did not treat the case of $_\odot_$ and i. Using the fact that $_\odot_$ is associative, and iterating the previous isomorphisms (and their analogues in the left variable), it is enough to prove that a map built exclusively from i and $_\odot_$ is a fibration. But such a map is isomorphic to δi^n from some n > 0, which is assumed to be a fibration.

Remark 80. The only part of the proof relying crucially on that fact that we have a cubical notion of parametricity is the step where we prove without using stability of fibrations by $_\odot_$ that if f a fibration, so is $f \otimes a$.

Now we want to analyse cofreely parametric models for cubical notions of parametricity.

Definition 81. Let C be a clan, and \Box_c be a cubical notion of parametricity generated by \Box .

• A functor $X : C^{\Box}$ is called fibrant if for all $n \ge 0$ we have an induced fibration:

$$X(\mathbb{I}^n) \twoheadrightarrow \overline{X}(\delta \mathbb{I}^n) \tag{229}$$

where \overline{X} extends X by commuting with limits.

• A morphism between:

$$X, Y: \mathcal{C}^{\square} \tag{230}$$

with X and Y fibrant is called a fibration if it is a pointwise fibration, and for all $n \ge 0$ we have fibrations:

$$X(\mathbb{I}^n) \twoheadrightarrow \overline{X}(\delta \mathbb{I}^n) \underset{\overline{Y}(\delta \mathbb{I}^n)}{\times} Y(\mathbb{I}^n)$$
 (231)

where \overline{X} (resp. \overline{Y}) extends X (resp. Y) by commuting with limits.

Remark 82. It should be noted that:

$$\overline{X}(\delta \mathbb{I}^n) \tag{232}$$

can be defined inductively by commuting with limits only if the maps:

$$X(\mathbb{I}^k) \to \overline{X}(\delta i^k) \tag{233}$$

are fibrations for k < n.

Remark 83. For \Box (the opposite of) a Reedy category of cubes where faces are the only dimension-decreasing maps, a fibration in the previous sense is precisely a Reedy fibration in C^{\Box} , and a fibrant object in C^{\Box} is precisely a Reedy fibrant object. This is the case for semi-cubes, or cubes with reflexivities only. On the other hand diagonals contradict this assumption. **Proposition 84.** Let C be a clan, and \Box_c be a cubical notion of parametricity generated by \Box . Then:

$$\Box_c \multimap \mathcal{C} \tag{234}$$

is the clan of fibrant objects in \mathcal{C}^{\Box} , equipped the fibrations from Definition 81.

Proof. By Lemma 79 giving an object in $\Box_c \multimap \mathcal{C}$ (i.e. a morphism of clan from \Box_c to \mathcal{C}) is the same as giving a fibrant object in \mathcal{C}^{\Box} .

Next we want to extend this to show that a fibration in $\Box_c \multimap \mathcal{C}$ is a fibration in the sense of Definition 81. We define Δ as the free clan generated by two objects and a fibration between them. For \mathcal{C} a clan we have that $\Delta \multimap \mathcal{D}$ is the clan 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:

$$\operatorname{Hom}_{\operatorname{Clan}}(\Delta, \Box_c \multimap \mathcal{C}) \cong \operatorname{Hom}_{\operatorname{Clan}}(\Box_c, \Delta \multimap \mathcal{C})$$
(235)

so that, by Lemma 79, a fibration in $\Box_c \multimap \mathcal{C}$ between X and Y is the same thing as:

- A functor from \Box to $\Delta \multimap C$ with source X and target Y. This is equivalent to a natural transformation α from X to Y restricted to \Box .
- Such that for any $n \ge 0$ we have that:

$$\begin{array}{c} X(\mathbb{I}^n) \xrightarrow{\alpha(\mathbb{I}^n)} Y(\mathbb{I}^n) \\ \downarrow \\ \chi(\delta\mathbb{I}^n) \xrightarrow{\overline{\alpha(\delta\mathbb{I}^n)}} Y(\delta\mathbb{I}^n) \end{array}$$

is Reedy fibrant, where $\overline{\alpha}$ extends α by commuting with limits.

But this is precisely a fibration in the sense of Definition 81.

Remark 85. This allows to generate clans of Reedy fibrant semi-cubical and cubical objects. We conjecture that this result can be extended to all Reedy category with a compatible monoidal structure, using the monoidal clan \Box with dimension-decreasing maps as fibrations. It is unclear to us what compatible should mean here.

Conclusion

First we give a summary of the presented a result, and then possible directions for future work.

Summary

We showed that notions of parametricity can be defined as monoidal models of type theory, and parametric models as modules. In one diagram:



We elaborate a bit:

- Assuming a symmetric monoidal closed category of models of type theory, a notion of parametricity was defined as a monoid □ (i.e. a monoidal model of type theory). Then a parametric model was defined as a model of type theory with a □-action, i.e. a □-module.
- We gave explicit formulas for the left and right adjoints to the functor forgetting a module structure, yielding freely parametric models and cubical models.
- We gave symmetric monoidal closed structures on the categories of categories, lex categories and clans, all of them seen as models of type theory.
- We gave many examples of parametricity, explaining how to construct the following as cofreely parametric models of type theory:
 - Categories of cubical objects (for any variant of cubes).
 - Lex categories of truncated cubical objects (for semi-cubes and cubes with reflexivities).
 - Clans of Reedy fibrant cubical objects (for semi-cubes and cubes with reflexivities).

Further work

We list several limitations of this work, and offer directions to deal with them.

• An obvious technical limitation is the requirement of strict commutativity of limits in lex categories and clans. The proper way to remove this is to work with a 2-category of models of type theory, removing the need for any strictness assumption.

An alternative, less satisfying approach would be to extend well-known strictification results for monoidal categories to monoidal lex categories and monoidal clans.

• We should construct symmetric monoidal closed structures on many other categories of models of type theory, including categories with families and comprehension categories. We want to design techniques allowing to do this efficiently.

We conjecture that all these monoidal closed structure can be build as some variant of Day convolution, using the fact that the lex categories classifying models of type theory are monoidal (for reasonable notions of model of type theory with Σ and \top). This would imply that any model of type theory is a model internal to the category of models of type theory. This is analogous to the fact that any abelian group is an abelian group internal to the category of abelian groups.

• It is unclear how to include Π and \mathcal{U} in this context. Indeed even in the relatively simple case of categories we cannot extend parametricity to exponentials. This would imply that the counit sending any cubical object to its underlying object of points commutes with exponentials, which fails in the presence of reflexivities (although this is true for semi-cubes, as seen in [17]).

However one can prove that when a category \mathcal{D} has exponentials and enough limits, the category $\mathcal{D}^{\mathcal{C}}$ has exponentials for any category \mathcal{C} . Perhaps this can be extended to a result giving Π and \mathcal{U} in any cubical model, assuming Π and \mathcal{U} together with enough inductive types in the base model.

• We cannot build clans with Kan fibrations as cofreely parametric models. Indeed for any clans \Box and \mathcal{D} , the fibrations in $\Box \multimap \mathcal{D}$ are always defined by the requirement that some maps in \mathcal{D} are fibrations, and never by the requirement that some terms exist, but these terms seem necessary to model the lifting in a Kan fibration.

Nevertheless we think that Kan cubical models can be constructed in two steps: first construct cubical objects, then restrict the fibrations. We hope that the second step can be compactly formulated in our setting, even if it does not consist in adding a module structure. In particular we conjecture that there is a right adjoint to the forgetful functor:

{Parametric models with Kan liftings} \rightarrow {Parametric models}

The notion of *parametricity with Kan liftings* should be equivalent to univalence, making formal the following correspondence:

Relation	Parametricity	Semi-cubes
Equivalence	Univalence	Kan cubes

This idea was already explored by Altenkirch and Kaposi [2] without fully computing with univalence, and by Tabareau, Tanter and Sozeau [20] assuming a univalent universe to begin with.

References

- Jiří Adámek, J Adamek, J Rosicky, et al. Locally presentable and accessible categories. 1994.
- [2] Thorsten Altenkirch and Ambrus Kaposi. Towards a cubical type theory without an interval. *Leibniz International Proceedings in Informatics*, 2017.
- [3] Jean-Philippe Bernardy, Thierry Coquand, and Guilhem Moulin. A presheaf model of parametric type theory. *Electronic Notes in Theoreti*cal Computer Science, 2015.
- [4] Jean-Philippe Bernardy, Patrik Jansson, and Ross Paterson. Parametricity and dependent types. In Proceedings of the 15th ACM SIGPLAN international conference on Functional programming, 2010.
- [5] Martin Brandenburg. Tensor categorical foundations of algebraic geometry. arXiv preprint arXiv:1410.1716, 2014.
- [6] Kenneth S Brown. Abstract homotopy theory and generalized sheaf cohomology. Transactions of the American Mathematical Society, 1973.
- [7] Ulrik Buchholtz and Edward Morehouse. Varieties of cubical sets. In International Conference on Relational and Algebraic Methods in Computer Science, 2017.
- [8] John Cartmell. Generalised algebraic theories and contextual categories. Annals of pure and applied logic, 1986.
- [9] Evan Cavallo and Robert Harper. Internal parametricity for cubical type theory. In 28th EACSL Annual Conference on Computer Science Logic, 2020.
- [10] Peter Dybjer. Internal type theory. In International Workshop on Types for Proofs and Programs, 1995.
- [11] Martin Hofmann. Syntax and semantics of dependent types. In Extensional Constructs in Intensional Type Theory. 1997.
- [12] Patricia Johann and Kristina Sojakova. Cubical categories for higherdimensional parametricity. arXiv preprint arXiv:1701.06244, 2017.
- [13] André Joyal. Notes on clans and tribes. arXiv preprint arXiv:1710.10238, 2017.
- [14] Chantal Keller and Marc Lasson. Parametricity in an impredicative sort. Computer Science Logic, 2012.
- [15] 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.

- [16] Per Martin-Löf. An intuitionistic theory of types: Predicative part. In Studies in Logic and the Foundations of Mathematics. 1975.
- [17] Hugo Moeneclaey. Parametricity and semi-cubical types. In 36th Annual ACM/IEEE Symposium on Logic in Computer Science, 2021.
- [18] Daniel G Quillen. Axiomatic homotopy theory. In Homotopical Algebra. 1967.
- [19] John C Reynolds. Types, abstraction and parametric polymorphism. In Proceedings of the IFIP 9th World Computer Congres, 1983.
- [20] Nicolas Tabareau, Éric Tanter, and Matthieu Sozeau. The marriage of univalence and parametricity. *Journal of the ACM (JACM)*, 2021.
- [21] Izumi Takeuti. The theory of parametricity in lambda cube (towards new interaction between category theory and proof theory). *Kyoto University Research Information Repository*, 2001.
- [22] Philip Wadler. Theorems for free! In Proceedings of the 4th international conference on Functional programming languages and computer architecture, 1989.