Parametricity and cubes

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

HoTTEST
21 October

Outline

Introduction

CwF of semi-cubical types

Categories of cubical objects

CwF of setoids

Clan of Reedy fibrant cubical objects

Tribes of Kan cubical objects

Conclusion

Presentation

Bio

PhD student on HoTT in Paris.

Collaborators:

- Hugo Herbelin (PhD advisor)
- Rafael Bocquet, Ambrus Kaposi (since march 2021)

Results presented here will be in my PhD dissertation.

Parametricity for type theory

Intuition

Polymorphic terms treats type input uniformly.

Parametricity for type theory

Intuition

Polymorphic terms treats type input uniformly.

- Types, abstraction and parametric polymorphism. [Reynolds 83]
- Theorems for free! [Wadler 89]
- Parametricity and dependent types. [Bernardy, Jansson, Paterson 10]

Cubical models

Intuition

Cubical structures are used to model parametricity and univalence.

Cubical models

Intuition

Cubical structures are used to model parametricity and univalence.

- A model of type theory in cubical sets.
  [Bezem, Coquand, Huber 14]
- Cubical type theory: a constructive interpretation of the univalence axiom. [Cohen, Coquand, Huber, Mörtberg 15]
- A presheaf model of parametric type theory.
  [Bernardy, Coquand, Moulin 15]
- Internal parametricity for cubical type theory.
  [Cavallo, Harper 20]

Univalence as a form of parametricity

- Towards a cubical type theory without an interval.
  [Altenkirch, Kaposi 15]
- The marriage of univalence and parametricity.
  [Tabareau, Tanter, Sozeau 20]

Parametric models

Intuition

A model of type theory is parametric if:

- Every type comes with a relation.
- Every term respects these.

Parametric models

Intuition

A model of type theory is parametric if:

- Every type comes with a relation.
- Every term respects these.

This implies that polymorphic terms treat type inputs uniformly.

Big picture

The forgetful functor:

$$\{\textit{Parametric models}\} \to \{\textit{Models of type theory}\}$$

tend to have a right adjoint, building cubical models.

Big picture

The forgetful functor:

$$\{\textit{Parametric models}\} \to \{\textit{Models of type theory}\}$$

tend to have a right adjoint, building cubical models.

In this talk

We get various cubical structures by using:

- Various notions of model of type theory.
- Various notions of parametricity.

A first example

Definition

The category $\square$ of semi-cubes is monoidal generated by:

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

$$d_0, d_1 : \mathbb{I} \to 1$$

A first example

Definition

The category $\square$ of semi-cubes is monoidal generated by:

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

$$d_0, d_1 : \mathbb{I} \to 1$$

A semi-cubical object in $\mathcal{C}$ is an object in $\mathcal{C}^{\square}$.

A first example

Definition

The category $\square$ of semi-cubes is monoidal generated by:

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

$$d_0, d_1 : \mathbb{I} \to 1$$

A semi-cubical object in $\mathcal{C}$ is an object in $\mathcal{C}^{\square}$.

Definition

A category is parametric if we are given:

- An endofunctor $\_{*}$.
- Two natural transformations:

$$0, 1 : X_{*} \to X$$

Theorem

The forgetful functor:

$$\{\textit{Parametric categories}\} \to \{\textit{Categories}\}$$

has a right adjoint:

$$\mathcal{C} \mapsto \mathcal{C}^{\square}$$

Outline

Introduction

CwF of semi-cubical types

Categories of cubical objects

CwF of setoids

Clan of Reedy fibrant cubical objects

Tribes of Kan cubical objects

Conclusion

Outline

Summary

Theorem [LICS 21]

The forgetful functor:

$$\{\textit{Parametric CwF with } \Pi, \mathcal{U}\} \to \{\textit{CwF with } \Pi, \mathcal{U}\}$$

has a right adjoint, building semi-cubical models.

Summary

Theorem [LICS 21]

The forgetful functor:

$$\{\textit{Parametric CwF with } \Pi, \mathcal{U}\} \to \{\textit{CwF with } \Pi, \mathcal{U}\}$$

has a right adjoint, building semi-cubical models.

In two steps:

- Axiomatize parametricity as an interpretation.
- Build a right adjoint from any interpretation.

Parametricity for type theory

We can define unary operations ($*$) inductively:

$$\begin{array}{ccc} \Gamma \vdash & \text{gives} & \Gamma_0, \Gamma_1 \vdash \Gamma_* \\ \Gamma \vdash A & \text{gives} & \Gamma_0, \Gamma_1, \Gamma_*, A_0, A_1 \vdash A_* \\ \Gamma \vdash a : A & \text{gives} & \Gamma_0, \Gamma_1, \Gamma_* \vdash a_* : A_*[a_0, a_1] \end{array}$$

Parametricity for type theory

We can define unary operations $(*)$ inductively:

$$
\begin{array}{rcc}
\Gamma \vdash & \text{gives} & \Gamma_0, \Gamma_1 \vdash \Gamma_* \\
\Gamma \vdash A & \text{gives} & \Gamma_0, \Gamma_1, \Gamma_*, A_0, A_1 \vdash A_* \\
\Gamma \vdash a : A & \text{gives} & \Gamma_0, \Gamma_1, \Gamma_* \vdash a_* : A_*[a_0, a_1]
\end{array}
$$

By equations (E) including:

$$
\begin{array}{rcl}
(A \times B)_*[(x_0, y_0), (x_1, y_1)] & = & A_*[x_0, x_1] \times B_*[y_0, y_1] \\
(A \to B)_*[\lambda x_0.t_0, \lambda x_1.t_1] & = & \Pi_{(x_0, x_1 : A)}\ A_*[x_0, x_1] \to B_*[t_0, t_1] \\
\mathcal{U}_*[X_0, X_1] & = & X_0 \to X_1 \to \mathcal{U}
\end{array}
$$

Interpretation

Definition

A CwF is called parametric if it has:

- Operations $(*)$
- Obeying equations (E)

Interpretation

Definition

A CwF is called parametric if it has:

- Operations $(*)$
- Obeying equations (E)

The initial CwF is parametric.

Definition [LICS 21]

An extension of the theory of CwF by:

- A family of unary operations.
- Equations defining them inductively.

is called an interpretation of CwF.

Definition [LICS 21]

An extension of the theory of CwF by:

- A family of unary operations.
- Equations defining them inductively.

is called an interpretation of CwF.

Parametricity is an interpretation of CwF.

Definition [LICS 21]

An extension of the theory of CwF by:

- A family of unary operations.
- Equations defining them inductively.

is called an interpretation of CwF.

Parametricity is an interpretation of CwF.

Theorem

The functor forgetting an interpretation has a right adjoint.

The right adjoint

Assume an interpretation of CwF by $(*)$ and (E). Then:

$$U : \{CwF + (*) + (E)\} \to \{CwF\}$$

has a right adjoint:

$$R : \{CwF\} \to \{CwF + (*) + (E)\}$$

The right adjoint

Assume an interpretation of CwF by $(*)$ and (E). Then:

$$U : \{CwF + (*) + (E)\} \to \{CwF\}$$

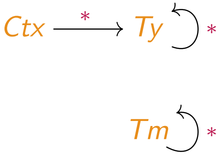
has a right adjoint:

$$R : \{CwF\} \to \{CwF + (*) + (E)\}$$

Intuition

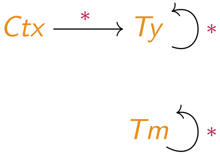
- A type in $R(\mathcal{C})$ is a type in $\mathcal{C}$ with iterated images by $(*)$.
- Same for contexts and terms.
- Operations in $R(\mathcal{C})$ are defined using operations in $\mathcal{C}$ and (E).

Example:



A type in $R(\mathcal{C})$ is:

Example:



A type in $R(\mathcal{C})$ is:

$$\vdash_{\mathcal{C}} \Gamma$$

$$\Gamma_0, \Gamma_1 \vdash_{\mathcal{C}} \Gamma_*$$

$$\Gamma_{00}, \Gamma_{01}, \Gamma_{0*}, \Gamma_{10}, \Gamma_{11}, \Gamma_{1*}, \Gamma_{*0}, \Gamma_{*1} \\ \vdash_{\mathcal{C}} \Gamma_{**}$$

$$\ldots$$

Example:

$$Ctx \xrightarrow{\ *\ } Ty \circlearrowleft *$$

$$Tm \circlearrowleft *$$

A type in $R(\mathcal{C})$ is:	A cubical type is:
$\vdash_{\mathcal{C}} \Gamma$	A type of points
$\Gamma_0, \Gamma_1 \vdash_{\mathcal{C}} \Gamma_*$	For any two points, a type of paths.
$\Gamma_{00}, \Gamma_{01}, \Gamma_{0*}, \Gamma_{10}, \Gamma_{11}, \Gamma_{1*}, \Gamma_{*0}, \Gamma_{*1} \vdash_{\mathcal{C}} \Gamma_{**}$	For any square, a type of fillers.
$\ldots$	$\ldots$

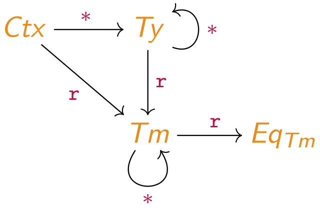
We can add reflexivities (when there is no Π or $\mathcal{U}$):

$$\begin{array}{ccc} \Gamma \vdash & \text{gives} & \Gamma \vdash \mathtt{r}_\Gamma : \Gamma_*[\gamma, \gamma] \\ \Gamma \vdash A & \text{gives} & \Gamma, A \vdash \mathtt{r}_A : A_*[r_\Gamma, a, a] \\ \Gamma \vdash a : A & \text{gives} & a_*[\mathtt{r}_\Gamma] = \mathtt{r}_A[a] \end{array}$$

We can add reflexivities (when there is no Π or $\mathcal{U}$):

$$\begin{array}{ccc} \Gamma \vdash & \text{gives} & \Gamma \vdash \mathtt{r}_\Gamma : \Gamma_*[\gamma, \gamma] \\ \Gamma \vdash A & \text{gives} & \Gamma, A \vdash \mathtt{r}_A : A_*[r_\Gamma, a, a] \\ \Gamma \vdash a : A & \text{gives} & a_*[\mathtt{r}_\Gamma] = \mathtt{r}_A[a] \end{array}$$

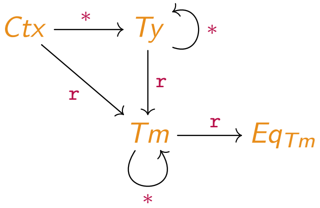
As represented:



We can add reflexivities (when there is no Π or $\mathcal{U}$):

$$\begin{array}{ccc} \Gamma \vdash & \text{gives} & \Gamma \vdash \mathtt{r}_\Gamma : \Gamma_*[\gamma, \gamma] \\ \Gamma \vdash A & \text{gives} & \Gamma, A \vdash \mathtt{r}_A : A_*[r_\Gamma, a, a] \\ \Gamma \vdash a : A & \text{gives} & a_*[\mathtt{r}_\Gamma] = \mathtt{r}_A[a] \end{array}$$

As represented:



A type in the new CwF is then a sequence $(A_{*n})_{n:\mathbb{N}}$ with:

$$\left((\mathtt{r}_{A_{*m}})_{*n}\right)_{m,n:\mathbb{N}}$$

obeying some equations.

This approach is very modular:

- In the notion of model of type theory.
- In the unary operations added.

This approach is very modular:

- In the notion of model of type theory.
- In the unary operations added.

Example

To add $\mathbb{N}$, it is enough to define:

$$
\begin{aligned}
\mathbb{N}_* &= Eq_{\mathbb{N}} : \mathbb{N} \to \mathbb{N} \to \mathcal{U} \\
0_* &= \_ : Eq_{\mathbb{N}}(0,0) \\
s_* &= \_ : Eq_{\mathbb{N}}(m,n) \to Eq_{\mathbb{N}}(m+1,n+1) \\
ind_*^{\mathbb{N}} &= \_ : \_
\end{aligned}
$$

Negative result

Problem

We can't define:

$$\mathtt{r}_{A \to B} \stackrel{?}{=} \phi(\mathtt{r}_A, \mathtt{r}_B)$$

Negative result

Problem

We can't define:

$$\mathtt{r}_{A\to B} \stackrel{?}{=} \phi(\mathtt{r}_A, \mathtt{r}_B)$$

Reflexivities are not part of an interpretation for exponentials.

Negative result

Problem

We can't define:

$$\mathtt{r}_{A \to B} \stackrel{?}{=} \phi(\mathtt{r}_A, \mathtt{r}_B)$$

Reflexivities are not part of an interpretation for exponentials.

Intuition

- Exponentials of cubical objects are not computed pointwise.
- Interpretations compute constructors pointwise.

Negative result

Problem

We can't define:

$$\mathtt{r}_{A\to B} \stackrel{?}{=} \phi(\mathtt{r}_A, \mathtt{r}_B)$$

Reflexivities are not part of an interpretation for exponentials.

Intuition

- Exponentials of cubical objects are not computed pointwise.
- Interpretations compute constructors pointwise.

From now on we forget about exponentials and universes.

Outline

General parametricity for categories

Goal

We want to define various parametricities for categories.

General parametricity for categories

Goal

We want to define various parametricities for categories.

Definition

A notion of parametricity for categories is

General parametricity for categories

Goal

We want to define various parametricities for categories.

Definition

A notion of parametricity for categories is a monoidal category $\square$.

General parametricity for categories

Goal

We want to define various parametricities for categories.

Definition

A notion of parametricity for categories is a monoidal category $\square$.

Definition

A category $\mathcal{C}$ is $\square$-parametric if we are given a monoidal functor:

$$\square \to End(\mathcal{C})$$

General parametricity for categories

Goal

We want to define various parametricities for categories.

Definition

A notion of parametricity for categories is a monoidal category $\square$.

Definition

A category $\mathcal{C}$ is $\square$-parametric if we are given a monoidal functor:

$$\square \to End(\mathcal{C})$$

This is precisely an action of monoid in $\{Categories\}$.

Examples

Semi-cubes

The category of semi-cubes is monoidal generated by:

$$d_0, d_1 \quad : \quad \mathbb{I} \to 1$$

Examples

Semi-cubes

The category of semi-cubes is monoidal generated by:

$$d_0, d_1 \quad : \quad \mathbb{I} \to 1$$

So a parametric category has natural transformations:

$$0, 1 \quad : \quad X_* \to X$$

Cubes

The category of cubes is monoidal generated by:

$$
\begin{aligned}
d_0, d_1 &: \mathbb{I} \to 1 \\
\mathtt{r} &: 1 \to \mathbb{I} \\
d_0 \circ \mathtt{r} &= id_1 \\
d_1 \circ \mathtt{r} &= id_1
\end{aligned}
$$

Cubes

The category of cubes is monoidal generated by:

$$\begin{aligned} d_0, d_1 &: \mathbb{I} \to 1 \\ \mathtt{r} &: 1 \to \mathbb{I} \\ d_0 \circ \mathtt{r} &= id_1 \\ d_1 \circ \mathtt{r} &= id_1 \end{aligned}$$

The corresponding parametricity is called internal.

Cubes

The category of cubes is monoidal generated by:

$$\begin{aligned} d_0, d_1 &: \mathbb{I} \to 1 \\ \mathtt{r} &: 1 \to \mathbb{I} \\ d_0 \circ \mathtt{r} &= id_1 \\ d_1 \circ \mathtt{r} &= id_1 \end{aligned}$$

The corresponding parametricity is called internal.

Varieties of cubes

All cube categories in [Bucholtz, Morehouse 17] are monoidal.

Main result

Let $\square$ be a monoidal category.

Theorem

The forgetful functor:

$$\{\square\text{–}\mathit{Parametric\ categories}\} \to \{\mathit{Categories}\}$$

has a right adjoint:

$$\mathcal{C} \mapsto \mathcal{C}^{\square}$$

Proof

Let M be a monoid in a cartesian closed category $\mathcal{C}$.

Proof

Let M be a monoid in a cartesian closed category $\mathcal{C}$.

Lemma

The forgetful functor:

$$\{M\text{-}action\} \to \mathcal{C}$$

has a right adjoint:

$$X \mapsto X^M$$

Proof

Let M be a monoid in a cartesian closed category $\mathcal{C}$.

Lemma

The forgetful functor:

$$\{M\text{–}action\} \to \mathcal{C}$$

has a right adjoint:

$$X \mapsto X^M$$

Proved using simply typed λ-calculus.

Proof using interpretations

Theorem

□-parametricity is an interpretation of categories.

Proof using interpretations

Theorem

□-parametricity is an interpretation of categories.

Straightforward assuming a presentation:

- Functors are inductively defined on morphisms.
- Naturality is inductively provable on morphisms.
- ...

Proof using interpretations

Theorem

$\square$-parametricity is an interpretation of categories.

Straightforward assuming a presentation:

- Functors are inductively defined on morphisms.
- Naturality is inductively provable on morphisms.
- ...

Corollary

The sequences build by interpretations are cubical objects.

Outline

Basic framework

We start from a type theory with two notions of types:

Sets	$\Gamma \vdash_S A$
Propositions	$\Gamma \vdash_P A$

With $\top$ and Σ for propositions (and possibly for sets).

Basic framework

We start from a type theory with two notions of types:

Sets	$\Gamma \vdash_S A$
Propositions	$\Gamma \vdash_P A$

With $\top$ and Σ for propositions (and possibly for sets).

Definition

The canonical model is such that:

- $\Gamma \vdash$ means Γ set.
- $\Gamma \vdash_S A$ means A set over Γ.
- $\Gamma \vdash_P A$ means A a part of Γ.

Setoid type theory

We add operations $(*)$:

$$
\begin{array}{lcc}
\Gamma \vdash & \text{gives} & \Gamma_0, \Gamma_1 \vdash_P \Gamma_* \\
 & \text{and} & \Gamma \vdash \mathtt{r}_\Gamma : \Gamma_* \\
\\
\Gamma \vdash_S A & \text{gives} & \Gamma_0, \Gamma_1, \Gamma_*, A_0, A_1 \vdash_P A_* \\
 & \text{and} & \Gamma, A \vdash \mathtt{r}_A : A_*[\mathtt{r}_\Gamma] \\
\\
\Gamma \vdash_P A & \text{gives} & \Gamma_0, \Gamma_1, \Gamma_*, A_0 \vdash \overrightarrow{coe}_A : A_1 \\
 & \text{and} & \Gamma_0, \Gamma_1, \Gamma_*, A_1 \vdash \overleftarrow{coe}_A : A_0
\end{array}
$$

Setoid type theory

We add operations $(*)$:

$$
\begin{array}{lcc}
\Gamma \vdash & \text{gives} & \Gamma_0, \Gamma_1 \vdash_P \Gamma_* \\
 & \text{and} & \Gamma \vdash \mathtt{r}_\Gamma : \Gamma_* \\
 & & \\
\Gamma \vdash_S A & \text{gives} & \Gamma_0, \Gamma_1, \Gamma_*, A_0, A_1 \vdash_P A_* \\
 & \text{and} & \Gamma, A \vdash \mathtt{r}_A : A_*[\mathtt{r}_\Gamma] \\
 & & \\
\Gamma \vdash_P A & \text{gives} & \Gamma_0, \Gamma_1, \Gamma_*, A_0 \vdash \overrightarrow{coe}_A : A_1 \\
 & \text{and} & \Gamma_0, \Gamma_1, \Gamma_*, A_1 \vdash \overleftarrow{coe}_A : A_0
\end{array}
$$

Plus equations defining $(*)$ inductively, notably for $\Gamma \vdash_P A$ we add:

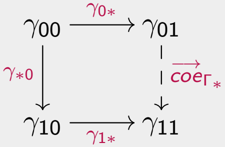
$$(\Gamma, A)_* = \Gamma_*$$

Remark

We have:

$$\Gamma_{00}, \Gamma_{10}, \Gamma_{01}, \Gamma_{11}, \Gamma_{0*}, \Gamma_{1*}, \Gamma_{*0} \vdash \overrightarrow{coe}_{\Gamma_*} : \Gamma_{*1}$$

In diagram:



So that Γ_* is reflexive, symmetric and transitive.

Remark

We have:

$$\Gamma_{00}, \Gamma_{10}, \Gamma_{01}, \Gamma_{11}, \Gamma_{0*}, \Gamma_{1*}, \Gamma_{*0} \vdash \overrightarrow{coe}_{\Gamma_*} : \Gamma_{*1}$$

In diagram:

$$\begin{array}{ccc} \gamma 00 & \xrightarrow{\gamma_{0*}} & \gamma 01 \\ {\scriptstyle \gamma_{*0}} \downarrow & & \downarrow {\scriptstyle \overrightarrow{coe}_{\Gamma_*}} \\ \gamma 10 & \xrightarrow[\gamma_{1*}]{} & \gamma 11 \end{array}$$

So that Γ_* is reflexive, symmetric and transitive.

Corollary

The canonical model is send to a model where:

- $\Gamma \vdash$ means Γ setoid.
- $\Gamma \vdash_S A$ means A setoid over Γ.
- $\Gamma \vdash_P A$ means A part of Γ stable by the relation.

Adding set transport

We can add operations:

$$\Gamma \vdash_S A \quad \text{gives} \quad \Gamma_0, \Gamma_1, \Gamma_*, A_0 \vdash \overrightarrow{coe}_A : A_1$$
$$\text{and} \quad \Gamma_0, \Gamma_1, \Gamma_*, A_1 \vdash \overleftarrow{coe}_A : A_0$$

with the equations:

$$\overrightarrow{coe}_A[\mathtt{r}_\Gamma, x] = x$$
$$\overleftarrow{coe}_A[\mathtt{r}_\Gamma, x] = x$$

Adding set transport

We can add operations:

$$\Gamma \vdash_S A \quad \text{gives} \quad \Gamma_0, \Gamma_1, \Gamma_*, A_0 \vdash \overrightarrow{coe}_A : A_1$$
$$\text{and} \quad \Gamma_0, \Gamma_1, \Gamma_*, A_1 \vdash \overleftarrow{coe}_A : A_0$$

with the equations:

$$\overrightarrow{coe}_A[\mathtt{r}_\Gamma, x] = x$$
$$\overleftarrow{coe}_A[\mathtt{r}_\Gamma, x] = x$$

This implies:

$$\overrightarrow{coh}_A : A_*[x_0, \overrightarrow{coe}_A(x_0)]$$
$$\overleftarrow{coh}_A : A_*[\overleftarrow{coe}_A(x_1), x_1]$$

Lemma

The canonical model is send to a model where:

- $\Gamma \vdash_S A$ means A fibration of setoid over Γ.

Lemma

The canonical model is send to a model where:

- $\Gamma \vdash_S A$ means A fibration of setoid over Γ.

These fibrations have non-reflexive transports as structure.

Adding constructors to the base theory

We can add the following:

- Π for propositions, for example:

$$\overrightarrow{coe}_{A \to B}[f] \qquad = \qquad A_1 \xrightarrow{\overleftarrow{coe}_A} A_0 \xrightarrow{f} B_0 \xrightarrow{\overrightarrow{coe}_B} B_1$$

Adding constructors to the base theory

We can add the following:

- Π for propositions, for example:

$$\overrightarrow{coe}_{A \to B}[f] \quad = \quad A_1 \xrightarrow{\overleftarrow{coe}_A} A_0 \xrightarrow{f} B_0 \xrightarrow{\overrightarrow{coe}_B} B_1$$

- A universe of propositions, that is:

$$\begin{array}{rcl} & \vdash_S & \mathcal{U} \\ \mathcal{U} & \vdash_P & El \end{array}$$

  with equations including:

$$\begin{array}{rcl} \mathcal{U}_*[A, B] & = & A \leftrightarrow B \\ \mathtt{r}_{\mathcal{U}}[A] & = & (id_A, id_A) \\ \overrightarrow{coe}_{El}[e] & = & e.1 \\ \overleftarrow{coe}_{El}[e] & = & e.2 \end{array}$$

This was lucky! We can't add the following:

- Π types for sets.
- A universe of sets.

Remark on modularity

Interpretation approach modular on constructors and equations:

- Want $\vdash_S \mathbb{N}$. Define $x, y : \mathbb{N} \vdash_P Eq_{\mathbb{N}}$ inductively.
- Don't like $(\overrightarrow{coe}_A)_*$ derivable. Remove this redundancy.
- Want $\overrightarrow{coe}_A[p \circ q] = \overrightarrow{coe}_A[p] \circ \overrightarrow{coe}_A[q]$. Prove it inductively.
- Don't like $\overrightarrow{coe}_A[\mathtt{r}_\Gamma, x] = x$. Try $\overrightarrow{coh}_A : A_*[x, \overrightarrow{coe}_A(x)]$ instead.
- ...

It gives a straightforward first try to tackle any of these issues.

Outline

Reminder on clan

Definition [Joyal 17]

A clan consists of:

$\mathcal{C}$ a category	Contexts and substitutions
1 a terminal object	Empty context
F a class of morphisms	Types

such that:

F stable by isomorphism	$\top$
F stable by composition	Σ
F stable by pullback	$A[\sigma]$
F stable by $X \to 1$	Democratic

Parametric clans

We use semi-cubes.

Parametric clans

We use semi-cubes.

Definition

A clan is parametric if we have:

- An endofunctor $\_{*}$ with natural transformations:

$$0, 1 : X_* \to X$$

- Obeying the fibration rule:

$$\frac{X \twoheadrightarrow Y}{X_* \twoheadrightarrow (X_0 \times X_1) \prod_{Y_0 \times Y_1} Y_*}$$

Parametric clans

We use semi-cubes.

Definition

A clan is parametric if we have:

- An endofunctor $\_{*}$ with natural transformations:

$$0, 1 : X_* \to X$$

- Obeying the fibration rule:

$$\frac{X \twoheadrightarrow Y}{X_* \twoheadrightarrow (X_0 \times X_1) \prod_{Y_0 \times Y_1} Y_*}$$

Note that:

$$\frac{\_ : X \twoheadrightarrow 1}{(0, 1) : X_* \twoheadrightarrow X \times X}$$

Claim (in progress)

Assume $f : A \to B$ in $\mathcal{C}^{\square}$ for $\mathcal{C}$ a clan.
Starting from $f_0 : A_0 \twoheadrightarrow B_0$ and iterating the fibration rule:

$$\frac{X \twoheadrightarrow Y}{X_* \twoheadrightarrow (X \times X) \prod_{Y \times Y} Y_*}$$

we get that f is Reedy fibration.

Claim (in progress)

Assume $f : A \to B$ in $\mathcal{C}^{\square}$ for $\mathcal{C}$ a clan.
Starting from $f_0 : A_0 \twoheadrightarrow B_0$ and iterating the fibration rule:

$$\frac{X \twoheadrightarrow Y}{X_* \twoheadrightarrow (X \times X) \prod_{Y \times Y} Y_*}$$

we get that f is Reedy fibration.

Claim (in progress)

Parametricity is an interpretation of clans.

Claim (in progress)

Assume $f : A \to B$ in $\mathcal{C}^{\square}$ for $\mathcal{C}$ a clan.
Starting from $f_0 : A_0 \twoheadrightarrow B_0$ and iterating the fibration rule:

$$\frac{X \twoheadrightarrow Y}{X_* \twoheadrightarrow (X \times X) \prod_{Y \times Y} Y_*}$$

we get that f is Reedy fibration.

Claim (in progress)

Parametricity is an interpretation of clans.

Corollary

The right adjoint to the forgetful functor:

$$\{\textit{Parametric clans}\} \to \{\textit{Clans}\}$$

sends $\mathcal{C}$ to the clan of Reedy fibrant semi-cubical objects in $\mathcal{C}$.

Outline

Reminder on tribes

Definition

A map is called anodyne if it has the LLP against fibrations.

Reminder on tribes

Definition

A map is called anodyne if it has the LLP against fibrations.

Definition [Joyal 17]

A tribe is a clan where:

- Every map factors as an anodyne map followed by a fibration.
- Anodyne maps are stable by pullback.

Reminder on tribes

Definition

A map is called anodyne if it has the LLP against fibrations.

Definition [Joyal 17]

A tribe is a clan where:

- Every map factors as an anodyne map followed by a fibration.
- Anodyne maps are stable by pullback.

A tribe is a model of type theory with identity types:

$$X \rightarrowtail Id_X \twoheadrightarrow X \times X$$

Here reflexivity being anodyne is equivalent to path induction.

Kan clan

We start from $\square$ the category of symmetric cubes.

Kan clan

We start from $\square$ the category of symmetric cubes.

Definition

A clan is called Kan if it is:

- $\square$-parametric as a category.
- Obeying the fibration rule.
- Such that for $A \twoheadrightarrow \Gamma$ we have sections of:

$$A_* \twoheadrightarrow A[0]$$
$$A_* \twoheadrightarrow A[1]$$

Kan clan

We start from $\Box$ the category of symmetric cubes.

Definition

A clan is called Kan if it is:

- $\Box$-parametric as a category.
- Obeying the fibration rule.
- Such that for $A \twoheadrightarrow \Gamma$ we have sections of:

$$A_* \twoheadrightarrow A[0]$$
$$A_* \twoheadrightarrow A[1]$$

A section of $A_* \twoheadrightarrow A[0]$ corresponds to $\overrightarrow{coe}_A$ and $\overrightarrow{coh}_A$ for setoids.

Theorem

A Kan clan is a tribe.

Theorem

A Kan clan is a tribe.

Proof:

- Factorisation for diagonals:

$$X \xrightarrow{\texttt{r}} X_* \xrightarrow{(0,1)} X \times X$$

- Coherences + Symmetry $\Rightarrow$ Contractibility of singletons.
- Contractibility of singletons + Coercions $\Rightarrow$ $\texttt{r}$ anodyne.

Theorem

A Kan clan is a tribe.

Proof:

- Factorisation for diagonals:

$$X \xrightarrow{\ \mathtt{r}\ } X_* \xrightarrow{\ (0,1)\ } X \times X$$

- Coherences + Symmetry $\Rightarrow$ Contractibility of singletons.
- Contractibility of singletons + Coercions $\Rightarrow$ $\mathtt{r}$ anodyne.
- Factorisation for a map f similar:

$$X \longrightarrow \Sigma_{x:X,y:Y}\, Y_*[f(x), y] \longrightarrow Y$$

Claim (in progress)

Being Kan is an interpretation of clans.

Claim (in progress)

Being Kan is an interpretation of clans.

Claim (in progress)

The associated right adjoint build tribes of Kan cubical objects.

Claim (in progress)

Being Kan is an interpretation of clans.

Claim (in progress)

The associated right adjoint build tribes of Kan cubical objects.

Sketch:

- $\overrightarrow{coh}_{\Gamma_{*n}}$ and $\overleftarrow{coh}_{\Gamma_{*n}}$ gives two Kan fillings per dimension.
- Symmetry gives all other Kan fillings.

Outline

Summary

Cubical models = Cofreely parametric models.

Summary

Cubical models = Cofreely parametric models. Examples:

- CwF of semi-cubical types, with Π and $\mathcal{U}$.
- Categories of cubical objects, for any kind of cubes.
- CwF of setoids.
- Clan of Reedy fibrant cubical objects (in progress).
- Tribes of Kan cubical objects (in progress).

Summary

Cubical models = Cofreely parametric models. Examples:

- CwF of semi-cubical types, with Π and $\mathcal{U}$.
- Categories of cubical objects, for any kind of cubes.
- CwF of setoids.
- Clan of Reedy fibrant cubical objects (in progress).
- Tribes of Kan cubical objects (in progress).

Relations	Parametricity	Semi-cubes
Reflexive relations	Internal parametricty	Cubes
...	...	...
Equivalences	Univalence	Kan cubes

Further Work

- Find general parametricity as interpretations for:
  - Lex categories and clans.
  - CwF and comprehension categories.

Further Work

- Find general parametricity as interpretations for:
  - Lex categories and clans.
  - CwF and comprehension categories.
- Generalize setoids to truncated cubical objects.

Further Work

- Find general parametricity as interpretations for:
  - Lex categories and clans.
  - CwF and comprehension categories.
- Generalize setoids to truncated cubical objects.
- Some work on inductive types:
  - Extend parametricity to inductive types.
  - Show any cubical model has higher inductive types.

Further Work

- Find general parametricity as interpretations for:
  - Lex categories and clans.
  - CwF and comprehension categories.
- Generalize setoids to truncated cubical objects.
- Some work on inductive types:
  - Extend parametricity to inductive types.
  - Show any cubical model has higher inductive types.
- Extend interpretations to deal with Π and $\mathcal{U}$.

Further Work

- Find general parametricity as interpretations for:
  - Lex categories and clans.
  - CwF and comprehension categories.
- Generalize setoids to truncated cubical objects.
- Some work on inductive types:
  - Extend parametricity to inductive types.
  - Show any cubical model has higher inductive types.
- Extend interpretations to deal with Π and $\mathcal{U}$.
- Make the link with cubical type theories by:
  - Studying syntactic cubical models as parametric.
  - Designing cubical calculi for any cubical model.