Tutorial on *Polynomial Functors and Type Theory*

Steve Awodey

Workshop on Polynomial Functors
Topos Institute
March 2022

Outline

1. Polynomials

Let $\mathcal{E}$ be a locally cartesian closed category.
Thus for every map $f : B \to A$ we have adjoint functors on the slice categories,

$$\begin{array}{c} B \\ \big\downarrow{\scriptstyle f} \\ A \end{array} \qquad\qquad \begin{array}{c} \mathcal{E}/B \\ \Sigma_f \;\big\downarrow \;\; \big\uparrow f^* \;\; \big\downarrow\; \Pi_f \\ \mathcal{E}/A \end{array}$$

When $A = 1$ we write

$$\Sigma_B \dashv B^* \dashv \Pi_B$$

for the corresponding functors determined by $B \to 1$.

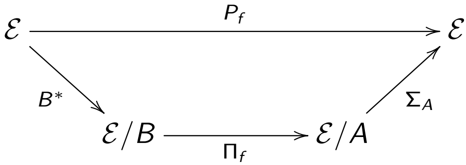
1. Polynomials

Definition

The *polynomial endofunctor* $P_f : \mathcal{E} \longrightarrow \mathcal{E}$ determined by a map

$$f : B \longrightarrow A$$

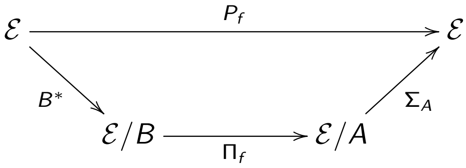
is the composite



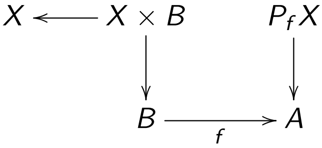
which we may write in the internal language of $\mathcal{E}$ as

$$\begin{aligned} P_f X = \Sigma_A \Pi_f B^* X &= \Sigma_A \Pi_f f^* A^* X \\ &= \Sigma_A \Pi_f f^* A^* X = \Sigma_A (A^* X)^f = \Sigma_{x:A} X^{B(x)}. \end{aligned}$$

1. Polynomials



The construction of $P_f X$ can be visualized as follows:



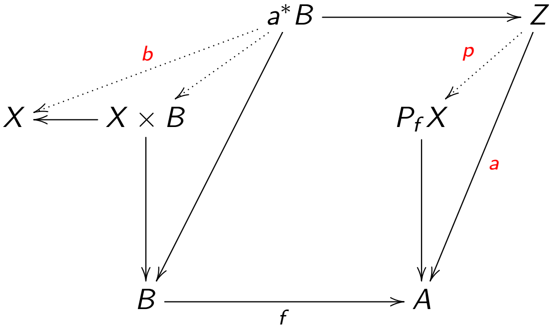
1. Polynomials

Lemma (UMP of $P_f X$)

Maps $p : Z \to P_f X$ correspond naturally to pairs (a, b) where

$$a : Z \to A \qquad b : a^* B \to X.$$

Proof.



∎

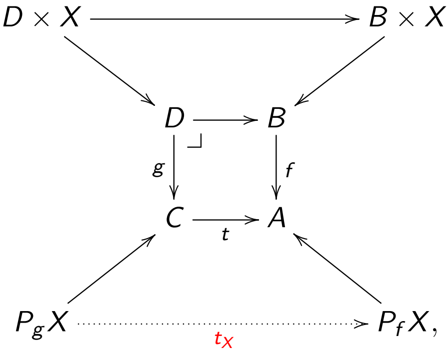
1. Polynomials

Now suppose we have a pullback square

$$\begin{array}{ccc} D & \longrightarrow & B \\ {\scriptstyle g}\downarrow & \lrcorner & \downarrow{\scriptstyle f} \\ C & \xrightarrow[t]{} & A. \end{array}$$

1. Polynomials

Then for each X we get a map $t_X : P_g X \to P_f X$ as follows:

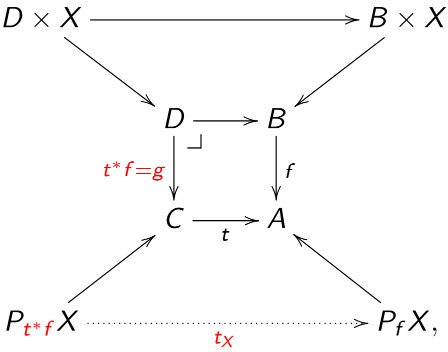


because the lower square is a pullback by Beck-Chavalley,

$$P_g X \cong t^* P_f X.$$

1. Polynomials

Then for each X we get map $t_X : P_g X \to P_f X$ as follows:



because the lower square is a pullback by Beck-Chavalley,

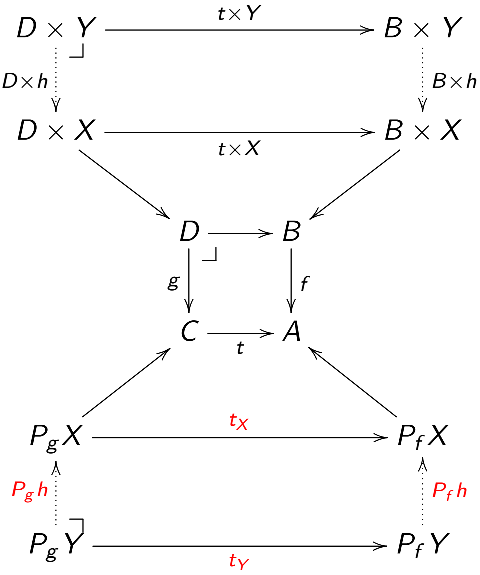
$$P_g X \cong t^* P_f X.$$

Indeed, since $g = t^* f$, we have

$$P_{t^* f} X \cong P_g X \cong t^* P_f X.$$

1. Polynomials

Then for each $h : Y \to X$ we have the pullback square below.



1. Polynomials

Proposition

Taking the polynomial functor $P_f : \mathcal{E} \to \mathcal{E}$ of a map $f : B \to A$ determines a functor

$$P : \mathcal{E}^{\to}_{\text{cart}} \longrightarrow \text{End}(\mathcal{E}).$$

The cartesian squares in $\mathcal{E}^{\to}$ are taken to cartesian natural transformations between endofunctors on $\mathcal{E}$. Moreover, the polynomials are closed under composition.

Proof.

It remains only to show that polynomial functors compose: given any $f : B \to A$ and $g : D \to C$, there is a map $h : F \to E$ such that

$$P_g \circ P_f = P_h : \mathcal{E} \longrightarrow \mathcal{E}.$$

See Spivak (2022) for the definition of $h = g \triangleleft f$. □

2. Dependent type theory

Types:

$$A, B, \ \ldots$$

Terms:

$$x : A, \ \ b : B, \ \ldots$$

Dependent types: ("type-indexed families of types")

$$x : A \vdash B(x)$$

$$x : A, y : B(x) \vdash C(x, y)$$

$$\ldots$$

Type forming operations:

$$\textstyle\sum_{x:A} B(x), \quad \prod_{x:A} B(x), \quad \ldots$$

Term forming operations:

$$\langle a, b \rangle, \quad \lambda x.b(x), \ \ldots$$

Equations:

$$s = t : A$$

2. Dependent type theory: Rules

Contexts:

$$\frac{x\!:\!A \vdash B(x)}{x\!:\!A,\ y\!:\!B(x) \vdash}$$

Writing Γ for any context, we have:

$$\frac{\Gamma \vdash C}{\Gamma, z\!:\!C \vdash}$$

2. Dependent type theory: Rules

Sums:

$$\frac{\Gamma, x : A \vdash B(x)}{\Gamma \vdash \sum_{x:A} B(x)} \qquad \frac{\Gamma \vdash a : A, \quad \Gamma \vdash b : B(a)}{\Gamma \vdash \langle a, b \rangle : \sum_{x:A} B(x)}$$

$$\frac{\Gamma \vdash c : \sum_{x:A} B(x)}{\Gamma \vdash \mathsf{fst}\, c : A} \qquad \frac{\Gamma \vdash c : \sum_{x:A} B(x)}{\Gamma \vdash \mathsf{snd}\, c : B(\mathsf{fst}\, c)}$$

$$\Gamma \vdash \mathsf{fst}\langle a, b \rangle = a : A \qquad \Gamma \vdash \mathsf{snd}\langle a, b \rangle = b : B$$

$$\Gamma \vdash \langle \mathsf{fst}\, c, \mathsf{snd}\, c \rangle = c : \sum_{x:A} B(x)$$

2. Dependent type theory: Rules

Sums:

$$\frac{\Gamma, x : A \vdash B(x)}{\Gamma \vdash \sum_{x:A} B(x)} \qquad \frac{\Gamma \vdash a : A, \quad \Gamma \vdash b : B(a)}{\Gamma \vdash \langle a, b \rangle : \sum_{x:A} B(x)}$$

$$\frac{\Gamma \vdash c : \sum_{x:A} B(x)}{\Gamma \vdash \mathsf{fst}\, c : A} \qquad \frac{\Gamma \vdash c : \sum_{x:A} B(x)}{\Gamma \vdash \mathsf{snd}\, c : B(\mathsf{fst}\, c)}$$

$$\Gamma \vdash \mathsf{fst}\langle a, b \rangle = a : A \qquad \Gamma \vdash \mathsf{snd}\langle a, b \rangle = b : B$$

$$\Gamma \vdash \langle \mathsf{fst}\, c, \mathsf{snd}\, c \rangle = c : \sum_{x:A} B(x)$$

2. Dependent type theory: Rules

Sums:

$$\frac{x : A \vdash B(x)}{\sum_{x:A} B(x)} \qquad \frac{a : A \qquad b : B(a)}{\langle a, b \rangle : \sum_{x:A} B(x)}$$

$$\frac{c : \sum_{x:A} B(x)}{\mathsf{fst}\, c : A} \qquad \frac{c : \sum_{x:A} B(x)}{\mathsf{snd}\, c : B(\mathsf{fst}\, c)}$$

$$\mathsf{fst}\langle a, b \rangle = a : A \qquad \mathsf{snd}\langle a, b \rangle = b : B$$

$$\langle \mathsf{fst}\, c, \mathsf{snd}\, c \rangle = c : \sum_{x:A} B(x)$$

2. Dependent type theory: Rules

Products:

$$\frac{x : A \vdash B(x)}{\prod_{x:A} B(x)} \qquad \frac{x : A \vdash b(x) : B(x)}{\lambda x.b(x) : \prod_{x:A} B(x)}$$

$$\frac{a : A \qquad f : \prod_{x:A} B(x)}{fa : B(a)}$$

$$x : A \vdash (\lambda x.b)x = b : B(x)$$

$$\lambda x.fx = f : \prod_{x:A} B(x)$$

2. Dependent type theory: Substitution

A tuple of terms in context $\sigma : \Delta \to \Gamma$ induces an operation

$$\frac{\sigma : \Delta \to \Gamma \qquad \Gamma \vdash a : A}{\Delta \vdash a[\sigma] : A[\sigma]}$$

which preserves *everything*.

For example given $y : Y \vdash s : Z$ and $z{:}Z, x{:}A(z) \vdash B(z,x)$ we can do

$$\frac{y : Y \vdash s : Z \qquad \frac{z:Z,\, x:A(z) \vdash B(z,x)}{z:Z \vdash \prod_{x:A(z)} B(z,x)}}{y : Y \vdash \left(\prod_{x:A(z)} B(z,x)\right)[s/z]} \quad \text{or} \quad \frac{\frac{y:Y \vdash s:Z \qquad z:Z,\, x:A(z) \vdash B(z,x)}{y:Y,\, x:A(s) \vdash B(s,x)}}{y : Y \vdash \prod_{x:A(s)} B(s,x)}$$

and syntactically the results are *the same*,

$$\left(\prod_{x:A(z)} B(z,x)\right)[s/z] = \prod_{x:A(s)} B(s,x)\,.$$

This suggests a reformulation as an *indexed algebraic structure*.

3. Natural models

Definition

A natural transformation $f : Y \to X$ of presheaves on a category $\mathbb{C}$ is called *representable* if its pullback along any $\mathrm{y}C \to X$ is representable:

$$\begin{array}{ccc} \mathrm{y}D & \longrightarrow & Y \\ \downarrow & \lrcorner & \downarrow{\scriptstyle f} \\ \mathrm{y}C & \longrightarrow & X \end{array}$$

Proposition (A, Fiore)

A representable natural transformation is the same thing as a **Category with Families** *in the sense of Dybjer.*

3. Natural models

Definition

A natural transformation $f : Y \to X$ of presheaves on a category $\mathbb{C}$ is called *representable* if its pullback along any $\mathrm{y}C \to X$ is representable: for all $C \in \mathbb{C}$ and $x \in X(C)$ there is given $p : D \to C$ and $y \in Y(D)$ such that the following is a pullback:

$$\begin{array}{ccc} \mathrm{y}D & \xrightarrow{\;y\;} & Y \\ {\scriptstyle \mathrm{y}p}\downarrow & \lrcorner & \downarrow{\scriptstyle f} \\ \mathrm{y}C & \xrightarrow[\;x\;]{} & X \end{array}$$

Proposition (A, Fiore)

A representable natural transformation equipped with a choice of such pullbacks is the same thing as a **Category with Families** *in the sense of Dybjer.*

3. Natural models

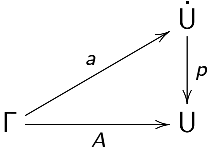
Write the objects and arrows of $\mathbb{C}$ as $\sigma : \Delta \to \Gamma$, thinking of a *category of contexts and substitutions*.

Let $p : \dot{\mathsf{U}} \to \mathsf{U}$ be a representable map of presheaves on $\mathbb{C}$.

Think of U as the *presheaf of types*, $\dot{\mathsf{U}}$ as the *presheaf of terms*, and then p gives the type of a term.

$$\Gamma \vdash A \quad \approx \quad A \in \mathsf{U}(\Gamma)$$
$$\Gamma \vdash a : A \quad \approx \quad a \in \dot{\mathsf{U}}(\Gamma)$$

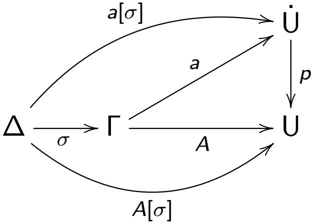
where $A = p \circ a$.



3. Natural models

Naturality of $p : \dot{\mathsf{U}} \to \mathsf{U}$ means that for any substitution $\sigma : \Delta \to \Gamma$, we have the required action on types and terms:

$$\Gamma \vdash A \quad \Rightarrow \quad \Delta \vdash A[\sigma]$$
$$\Gamma \vdash a : A \quad \Rightarrow \quad \Delta \vdash a[\sigma] : A[\sigma]$$



3. Natural models

Given any further $\tau : \Delta' \to \Delta$ we clearly have

$$A[\sigma][\tau] = A[\sigma \circ \tau] \qquad a[\sigma][\tau] = a[\sigma \circ \tau]$$

and for the identity substitution $1 : \Gamma \to \Gamma$

$$A[1] = A \qquad a[1] = a.$$

This is the basic structure of a CwF.

2. Natural models, context extension

The remaining operation of **context extension**

$$\frac{\Gamma \vdash A}{\Gamma, x : A \vdash}$$

is modeled by the representability of $p : \dot{\mathrm{U}} \to \mathrm{U}$ as follows.

3. Natural models, context extension

Given $\Gamma \vdash A$ we need a new context $\Gamma.A$ together with a substitution $p_A : \Gamma.A \to A$ and a term

$$\Gamma.A \vdash q_A : A[p_A]\,.$$

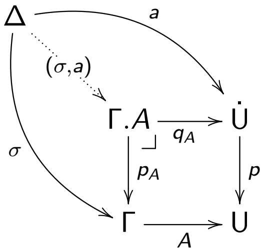
Let $p_A : \Gamma.A \to \Gamma$ be the pullback of p along A.

$$\begin{array}{ccc} \Gamma.A & \xrightarrow{q_A} & \dot{\mathrm{U}} \\ {\scriptstyle p_A}\downarrow & \lrcorner & \downarrow{\scriptstyle p} \\ \Gamma & \xrightarrow[A]{} & \mathrm{U} \end{array}$$

The map $q_A : \Gamma.A \to \dot{\mathrm{U}}$ gives the required term $\Gamma.A \vdash q_A : A[p_A]$. Syntactically, this is just the term

$$\Gamma, x : A \vdash x : A\,.$$

3. Natural models, context extension



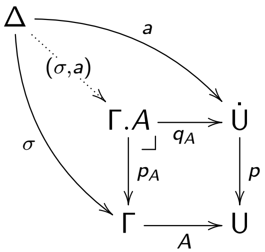
The pullback means that given any substitution $\sigma : \Delta \to \Gamma$ and term $\Delta \vdash a : A[\sigma]$ there is a map

$$(\sigma, a) : \Delta \to \Gamma.A$$

satisfying

$$p_A(\sigma, a) = \sigma$$
$$q_A[\sigma, a] = a.$$

3. Natural models, context extension



By the uniqueness of (σ, a), we also have

$$(\sigma, a) \circ \tau \;=\; (\sigma \circ \tau, a[\tau]) \qquad \text{for any } \tau : \Delta' \to \Delta$$

and

$$(p_A, q_A) = 1.$$

These are *all* the laws for a CwF. □

3. Natural models, algebraic formulation

Natural models can be presented as an essentially algebraic theory, with several sorts, partial operations, and equations between terms.

We have four basic sorts:

$$C_0,\ C_1,\ A,\ B$$

and the following operations and equations:

category: the usual domain, codomain, identity and composition operations for the index category $\mathbb{C}$:

$$C_1 \times_{C_0} C_1 \xrightarrow{\circ} C_1 \underset{\mathsf{dom}}{\overset{\mathsf{cod}}{\rightrightarrows}} C_0\,, \qquad C_1 \xleftarrow{\mathsf{id}} C_0$$

together with the familiar equations for a category.

3. Natural models, algebraic formulation

presheaf: the indexing and action operations for the presheaves $A, B : \mathbb{C}^{\mathsf{op}} \to \mathsf{Set}$:

$$\begin{array}{ccc} C_1 \times_{C_0} A & \xrightarrow{\alpha} & A \\ & & \downarrow{\scriptstyle p_A} \\ & & C_0 \end{array} \qquad \begin{array}{ccc} C_1 \times_{C_0} B & \xrightarrow{\beta} & B \\ & & \downarrow{\scriptstyle p_B} \\ & & C_0 \end{array}$$

together with the equations making α an action:

$$\begin{aligned} p_A(\alpha(u, a)) &= \mathrm{dom}(u), \\ \alpha(u \circ v, a) &= \alpha(v, \alpha(u, a)), \\ \alpha(1_{p_A(a)}, a) &= a, \end{aligned}$$

and similarly for β.

3. Natural models, algebraic formulation

natural transformation: an operation

$$f : A \to B$$

satisfying the naturality equations:

$$p_B \circ f = p_A, \qquad f \circ \alpha = \beta \circ (C_1 \times_{C_0} f).$$

representable: a natural transformation $f : A \to B$ is representable just if the associated functor,

$$\textstyle\int_{\mathbb{C}} f : \int_{\mathbb{C}} A \to \int_{\mathbb{C}} B$$

on the categories of elements has a right adjoint

$$f^* : \textstyle\int_{\mathbb{C}} B \to \int_{\mathbb{C}} A$$

(an algebraic condition, see Newstead (2018)).

3. Natural models and initiality

- The notion of a natural model is thus *essentially algebraic*.
- The algebraic homomorphisms correspond exactly to syntactic translations.
- There is an *initial algebra* as well as a *free algebra* over any signature of basic types and terms.
- The rules of dependent type theory specify a procedure for generating the free algebra.

3. Natural models and tribes

Let $p : \dot{\mathsf{U}} \to \mathsf{U}$ be a natural model.

The fibration

$$\textstyle\int_{\mathbb{C}} \mathsf{U} \to \mathbb{C}$$

of all *display maps* $p_A : \Gamma.A \to \Gamma$, for all $A : \Gamma \to \mathsf{U}$, determines a *clan* in the sense of Joyal (2017).

Conversely, given a clan $\mathcal{D} \hookrightarrow \mathbb{C}^{\to}$, there is a natural model in $\hat{\mathbb{C}}$,

$$\textstyle\coprod_{f \in \mathcal{D}} \mathsf{y}f : \coprod_{f \in \mathcal{D}} \mathsf{ydom}(f) \longrightarrow \coprod_{f \in \mathcal{D}} \mathsf{ycod}(f).$$

This natural model $p_{\mathcal{D}} : \dot{\mathsf{U}}_{\mathcal{D}} \to \mathsf{U}_{\mathcal{D}}$ determines a *splitting* of the associated fibration $\mathcal{D} \to \mathbb{C}$.

3. Natural models and tribes

Theorem (ish)

There is an adjunction between the categories of clans and of natural models, which specializes to a biequivalence between (certain) tribes and natural models with (certain) type-forming operations.

See A. (2017) for details.

Part II

4. Universes in presheaves

Recall the notion of a **Hofmann-Streicher universe**

$$\dot{\mathsf{V}} \to \mathsf{V}$$

in a category of presheaves $\widehat{\mathbb{C}} = \mathsf{Set}^{\mathbb{C}^{\mathsf{op}}}$.

1. Let set $\hookrightarrow$ Set be the full subcategory of *small* sets $s < \kappa$.
2. Let $\dot{\mathsf{set}} = 1/\mathsf{set}$ be the category of small *pointed* sets.
3. Then for $c \in \mathbb{C}$ let:

$$\mathsf{V}(c) = \mathsf{Cat}\big(\mathbb{C}/_c{}^{\mathsf{op}}, \mathsf{set}\big) \text{ the } \textit{set} \text{ of small presheaves on } \mathbb{C}/_c,$$
$$\dot{\mathsf{V}}(c) = \mathsf{Cat}\big(\mathbb{C}/_c{}^{\mathsf{op}}, \dot{\mathsf{set}}\big) \text{ ... small } \textit{pointed} \text{ presheaves on } \mathbb{C}/_c.$$

4. The action on $d \to c$ is given by *pre*composition with *post*composition $\mathbb{C}/_d \to \mathbb{C}/_c$.
5. There is a natural transformation $\dot{\mathsf{V}} \to \mathsf{V}$ determined by composing with the forgetful functor $\dot{\mathsf{set}} \to \mathsf{set}$

4. Universes in presheaves

Definition

In a category $\widehat{\mathbb{C}} = \mathsf{Set}^{\mathbb{C}^{op}}$ of presheaves,

- an object A is *small* if its values $A(c)$ are small, for all $c \in \mathbb{C}$,
- a map $A \to X$ is *small* if its fibers $A_x = x^*A$ are small, for all $x : \mathsf{y}c \to X$,

$$
\begin{array}{ccc}
A_x & \longrightarrow & A \\
\big\downarrow & \lrcorner & \big\downarrow \\
\mathsf{y}c & \xrightarrow[x]{} & X.
\end{array}
$$

Note that small maps are stable under pullback.
And that the map $\dot{\mathsf{V}} \to \mathsf{V}$ is small, since the fiber $\dot{\mathsf{V}}_S$ over $S : \mathsf{y}c \to \mathsf{V}$ has as elements pointed presheaves $\dot{S} : \mathbb{C}/_c \to \dot{\mathsf{set}}$.

4. Universes in presheaves

Proposition

For every small map $A \to X$ there is a canonical classifying map $\alpha : X \to \mathsf{V}$ fitting into a pullback diagram of the form

$$\begin{array}{ccc} A & \longrightarrow & \dot{\mathsf{V}} \\ \big\downarrow & \lrcorner & \big\downarrow \\ X & \xrightarrow[\alpha]{} & \mathsf{V}. \end{array}$$

Proof.

Do it first for the small maps $A_x \to \mathsf{y}c$, for all $x : \mathsf{y}c \to X$, for which there is a canonical choice of $\alpha_x : \mathsf{y}c \to \mathsf{V}$. Then use the presentation of X as a colimit over its category of elements $(c, x) \in \int_{\mathbb{C}} X$ to get $\alpha : X \to \mathsf{U}$. □

4. Universes in presheaves

Remark

For large enough κ the small maps are closed under the adjoints $\Sigma_A \dashv A^* \dashv \Pi_A$ to pullback along small maps $A \to X$.

This fact gives rise to natural operations on the universe $\dot{\mathsf{V}} \to \mathsf{V}$ that can be used to (coherently!) model the corresponding type-forming operations, as follows:

- a universe $\dot{\mathsf{V}} \to \mathsf{V}$ is a natural model on the category of contexts $\widehat{\mathbb{C}}$,
- a universe $\dot{\mathsf{V}} \to \mathsf{V}$ *generates* a polynomial endofunctor

$$P : \widehat{\mathbb{C}} \longrightarrow \widehat{\mathbb{C}}.$$

- The type forming operations in the natural model will correspond to algebraic structure on the polynomial endofunctor.

5. Polynomial monad and type formers

Let $p : \dot{\mathsf{U}} \to \mathsf{U}$ be a natural model on an arbitrary category $\mathbb{C}$, and consider the associated polynomial endofunctor,

$$P = \mathsf{U}_! \circ p_* \circ \dot{\mathsf{U}}^* : \widehat{\mathbb{C}} \longrightarrow \widehat{\mathbb{C}}\,,$$

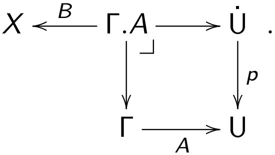
which we can write as,

$$P(X) \;=\; \sum_{A\,:\,\mathsf{U}} X^{[A]},$$

where $[A] = p^{-1}(A)$ is the fiber of $p : \dot{\mathsf{U}} \to \mathsf{U}$ at $A : \mathsf{U}$.

Lemma

Maps $\Gamma \to P(X)$ *correspond naturally to pairs* (A, B) *where*



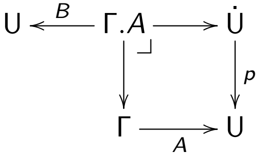
□

5. Polynomial monad and type formers

Applying P to U *itself* therefore gives the object

$$P\mathsf{U} = \sum_{A:\mathsf{U}} \mathsf{U}^{[A]},$$

for which maps $\Gamma \to P\mathsf{U}$ correspond naturally to pairs (A, B) of the form,



Since maps $\Gamma \to \mathsf{U}$ correspond naturally to types in context $\Gamma \vdash A$, we see that maps $\Gamma \to P\mathsf{U}$ correspond naturally to types in the extended context $\Gamma.A \vdash B$.

5. Polynomial monad and type formers

Proposition

For a natural model $\dot{\mathsf{U}} \to \mathsf{U}$, the polynomial object

$$P\mathsf{U} = \sum_{A:\mathsf{U}} \mathsf{U}^{[A]}$$

classifies types in context. Specifically, there is a natural isomorphism between maps $\Gamma \to P\mathsf{U}$ and pairs (A, B) where

$$\Gamma.A \vdash B.$$

Similarly, the object

$$P\dot{\mathsf{U}} = \sum_{A:\mathsf{U}} \dot{\mathsf{U}}^{[A]}$$

classifies *terms* in context: pairs $(A, b : B)$ where $\Gamma.A \vdash b : B$, for (A, B) the composite with $Pp : P\dot{\mathsf{U}} \to P\mathsf{U}$.

5. Polynomial monad and type formers: Π

Proposition

The natural model $p : \dot{\mathsf{U}} \to \mathsf{U}$ models the rules for products just if there are maps λ, Π making the following a pullback.

$$\begin{array}{ccc} P\dot{\mathsf{U}} & \xrightarrow{\lambda} & \dot{\mathsf{U}} \\ {\scriptstyle Pp}\downarrow & & \downarrow{\scriptstyle p} \\ P\mathsf{U} & \xrightarrow[\Pi]{} & \mathsf{U} \end{array}$$

5. Polynomial monad and type formers: Π

Proposition

The map $p : \dot{\mathsf{U}} \to \mathsf{U}$ *models the rules for products just if there are maps* λ, Π *making the following a pullback.*

Proof:

$$\sum_{A:\mathsf{U}} \mathsf{U}^{[A]}$$

$$\begin{array}{ccc} P(\dot{\mathsf{U}}) & \xrightarrow{\lambda} & \dot{\mathsf{U}} \\ \downarrow & & \downarrow \\ P(\mathsf{U}) & \xrightarrow[\Pi]{} & \mathsf{U} \end{array}$$

5. Polynomial monad and type formers: Π

Proposition

The map $p : \dot{\mathsf{U}} \to \mathsf{U}$ models the rules for products just if there are maps λ, Π making the following a pullback.

Proof:

$$\begin{array}{ccc} P(\dot{\mathsf{U}}) & \xrightarrow{\lambda} & \dot{\mathsf{U}} \\ \downarrow & & \downarrow \\ P(\mathsf{U}) & \xrightarrow[\Pi]{} & \mathsf{U} \end{array}$$

$$\sum_{A:\mathsf{U}} \mathsf{U}^{[A]}$$

$A \vdash B$ $\qquad\qquad$ $\Pi_A B$

5. Polynomial monad and type formers: Π

Proposition

The map $p : \dot{\mathsf{U}} \to \mathsf{U}$ models the rules for products just if there are maps λ, Π making the following a pullback.

Proof:

$$A \vdash b : B \qquad\qquad\qquad \lambda_A b$$

$$\begin{array}{ccccc} \sum_{A:\mathsf{U}} \dot{\mathsf{U}}^{[A]} & \quad & P(\dot{\mathsf{U}}) & \xrightarrow{\lambda} & \dot{\mathsf{U}} \\ & & \downarrow & & \downarrow \\ \sum_{A:\mathsf{U}} \mathsf{U}^{[A]} & \quad & P(\mathsf{U}) & \xrightarrow[\Pi]{} & \mathsf{U} \end{array}$$

$$A \vdash B \qquad\qquad\qquad \Pi_A B$$

5. Polynomial monad and type formers: Π

Proposition

The map $p : \dot{\mathsf{U}} \to \mathsf{U}$ *models the rules for products just if there are maps* λ, Π *making the following a pullback.*

Proof:

$$f$$

$$\begin{array}{cccc}
\sum_{A:\mathsf{U}} \dot{\mathsf{U}}^{[A]} & P(\dot{\mathsf{U}}) & \xrightarrow{\lambda} & \dot{\mathsf{U}} \\
 & \downarrow & & \downarrow \\
\sum_{A:\mathsf{U}} \mathsf{U}^{[A]} & P(\mathsf{U}) & \xrightarrow[\Pi]{} & \mathsf{U}
\end{array}$$

$$A \vdash B \qquad\qquad \Pi_A B$$

5. Polynomial monad and type formers: Π

Proposition

The map $p : \dot{\mathsf{U}} \to \mathsf{U}$ models the rules for products just if there are maps λ, Π making the following a pullback.

Proof:

$A \vdash fx : B$ $\qquad\qquad$ $\lambda_A fx = f$

$$\sum_{A:\mathsf{U}} \dot{\mathsf{U}}^{[A]} \qquad \begin{array}{ccc} P(\dot{\mathsf{U}}) & \xrightarrow{\ \lambda\ } & \dot{\mathsf{U}} \\ \downarrow & & \downarrow \\ P(\mathsf{U}) & \xrightarrow[\ \Pi\ ]{} & \mathsf{U} \end{array}$$

$$\sum_{A:\mathsf{U}} \mathsf{U}^{[A]}$$

$A \vdash B$ $\qquad\qquad$ $\Pi_A B$

5. Polynomial monad and type formers: Σ

Proposition

The map $p : \dot{\mathsf{U}} \to \mathsf{U}$ *models the rules for sums just if there are maps* (pair, Σ) *making the following a pullback*

$$\begin{array}{ccc} Q & \xrightarrow{\ \mathsf{pair}\ } & \dot{\mathsf{U}} \\ {\scriptstyle q}\downarrow & & \downarrow{\scriptstyle p} \\ P(\mathsf{U}) & \xrightarrow[\ \Sigma\ ]{} & \mathsf{U} \end{array}$$

where $q = p \triangleleft p : Q \to P(\mathsf{U})$ *is the generating map of the composite* $P_q = P_{p \triangleleft p} = P_p \circ P_p$.

Explicitly:

$$Q \;=\; \sum_{A:\mathsf{U}} \sum_{B:\mathsf{U}^A} \sum_{x:A} B(x)$$

5. Polynomial monad and type formers: T

Rules for a terminal type T

$$\frac{}{\vdash \mathsf{T}} \qquad \frac{}{\vdash * : \mathsf{T}} \qquad \frac{}{x : \mathsf{T} \vdash x = * : \mathsf{T}}$$

Proposition

The map $p : \dot{\mathsf{U}} \to \mathsf{U}$ models the rules for a terminal type just if there are maps $(*, \mathsf{T})$ making the following a pullback.

$$\begin{array}{ccc} 1 & \xrightarrow{\quad * \quad} & \dot{\mathsf{U}} \\ \downarrow & & \downarrow p \\ 1 & \xrightarrow[\quad \mathsf{T} \quad]{} & \mathsf{U} \end{array}$$

5. Polynomial monad

Consider the pullback squares for T and Σ.

$$
\begin{array}{ccc}
1 & \xrightarrow{\;*\;} & \dot{\mathsf{U}} \\
\downarrow & & \downarrow{\scriptstyle p} \\
1 & \xrightarrow[\mathsf{T}]{} & \mathsf{U}
\end{array}
\qquad\qquad
\begin{array}{ccc}
Q & \xrightarrow{\;\mathsf{pair}\;} & \dot{\mathsf{U}} \\
{\scriptstyle p \triangleleft p}\downarrow & & \downarrow{\scriptstyle p} \\
P(\mathsf{U}) & \xrightarrow[\Sigma]{} & \mathsf{U}
\end{array}
$$

These determine cartesian natural transformations between the corresponding polynomial endofunctors.

$$\tau : 1 \Rightarrow P \qquad\qquad \sigma : P \circ P \Rightarrow P$$

5. Polynomial monad

Theorem (A-Newstead)

A natural model $p : \dot{\mathsf{U}} \to \mathsf{U}$ models the T and Σ type formers just if the associated polynomial endofunctor P has the structure maps of a cartesian monad.

$$\tau : 1 \Rightarrow P \qquad\qquad \sigma : P \circ P \Rightarrow P$$

What about the monad laws?

5. Polynomial monad

The monad laws correspond to the following type isomorphisms.

$\sigma \circ P\sigma = \sigma \circ \sigma_P$	$\sum_{a:A} \sum_{b:B(a)} C(a,b) \;\cong\; \sum_{(a,b):\sum_{a:A} B(a)} C(a,b)$
$\sigma \circ P\tau = 1$	$\sum_{a:A} 1 \;\cong\; A$
$\sigma \circ \tau_P = 1$	$\sum_{x:1} A \;\cong\; A$

5. Polynomial monad

The pullback square for Π

$$\begin{array}{ccc} P\dot{\mathsf{U}} & \xrightarrow{\lambda} & \dot{\mathsf{U}} \\ {\scriptstyle Pp}\downarrow & & \downarrow{\scriptstyle p} \\ P\mathsf{U} & \xrightarrow[\Pi]{} & \mathsf{U} \end{array}$$

determines a cartesian natural transformation

$$\pi : P^{\mathbf{2}}p \Rightarrow p$$

where $P^{\mathbf{2}} : \hat{\mathbb{C}}^{\mathbf{2}} \to \hat{\mathbb{C}}^{\mathbf{2}}$ is the extension of P to the arrow category.

5. Polynomial monad

Theorem (A-Newstead)

A natural model $p : \dot{\mathsf{U}} \to \mathsf{U}$ *models the* Π *type former just if it has an algebra structure for the extended endofunctor* $P^{\mathbf{2}}$,

$$\pi : P^{\mathbf{2}} p \Rightarrow p.$$

5. Polynomial monad

The algebra laws correspond to the following type isomorphisms.

$\pi \circ P\pi \; = \; \pi \circ \sigma$	$\prod_{a:A} \prod_{b:B(a)} C(a,b) \; \cong \; \prod_{(a,b):\sum_{a:A} B(a)} C(a,b)$
$\pi \circ \tau \; = \; 1$	$\prod_{x:1} A \; \cong \; A$

6. Propositions and types

We can compare these operations on types

$$\Sigma, \Pi : P\mathsf{U} \longrightarrow \mathsf{U}$$

with those on subobjects of objects A in the topos $\widehat{\mathbb{C}}$,

$$\exists_A, \forall_A : \Omega^A \longrightarrow \Omega.$$

Consider

$$P\Omega \;=\; \sum_{A:\mathsf{U}} \Omega^A$$

for the polynomial endofunctor of $\dot{\mathsf{U}} \to \mathsf{U}$.
We then have the comparable maps

$$\exists, \forall : P\Omega \longrightarrow \Omega.$$

6. Propositions and types

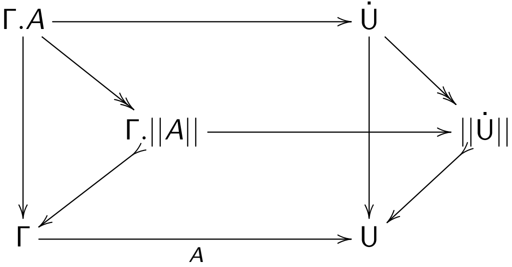
Proposition

There is a retraction $i : \Omega \rightarrowtail \mathsf{U}$, $s : \mathsf{U} \twoheadrightarrow \Omega$ *such that the following squares commute.*

$$\begin{CD} P\Omega @>{\exists}>> \Omega \\ @V{Pi}VV @AA{s}A \\ P\mathsf{U} @>>{\Sigma}> \mathsf{U} \end{CD} \qquad \qquad \begin{CD} P\Omega @>{\forall}>> \Omega \\ @V{Pi}VV @AA{s}A \\ P\mathsf{U} @>>{\Pi}> \mathsf{U} \end{CD}$$

6. Propositions and types

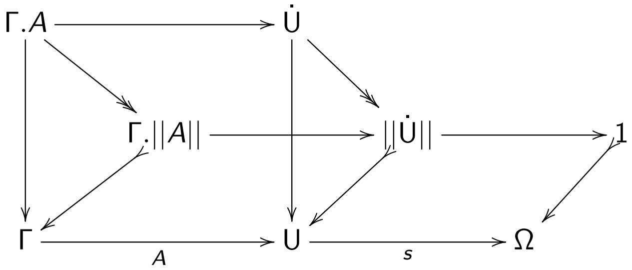
For the proof, factor the natural model $p : \dot{\mathsf{U}} \to \mathsf{U}$ as on the right below.



So $||\dot{\mathsf{U}}|| \rightarrowtail \mathsf{U}$ is a universal family of small propositions.

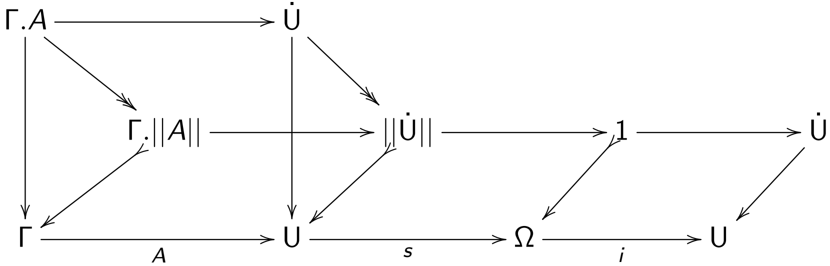
6. Propositions and types

Let $s : \mathsf{U} \to \Omega$ classify the mono $||\dot{\mathsf{U}}|| \rightarrowtail \mathsf{U}$.



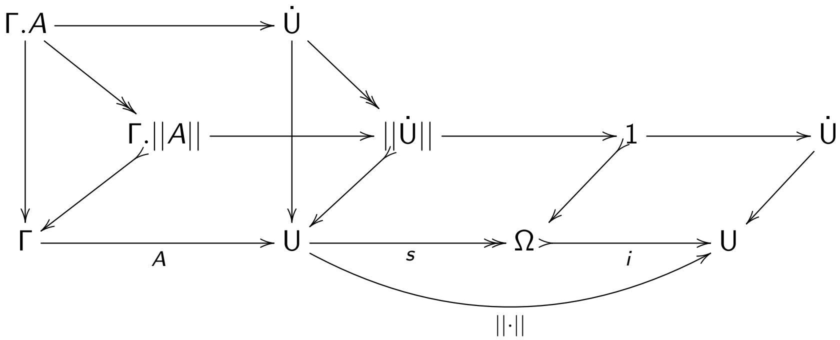
6. Propositions and types

Let $s : \mathsf{U} \to \Omega$ classify the mono $||\dot{\mathsf{U}}|| \rightarrowtail \mathsf{U}$.



Let $i : \Omega \to \mathsf{U}$ classify the family of small propositions $1 \rightarrowtail \Omega$.

6. Propositions and types



Let

$$||\cdot|| := i \circ s : \mathsf{U} \to \mathsf{U}.$$

We have

$$s \circ i = 1 : \Omega \to \Omega.$$

So

$$\Omega = \mathrm{im}(||\cdot||).$$

6. Propositions and types

The following diagrams then commute, as required.

$$
\begin{array}{ccc}
\sum\limits_{A:\mathsf{U}} \Omega^A & \xrightarrow{\quad\exists\quad} & \Omega \\
{\scriptstyle Pi}\downarrow & & \uparrow{\scriptstyle s} \\
\sum\limits_{A:\mathsf{U}} \mathsf{U}^A & \xrightarrow[\quad\Sigma\quad]{} & \mathsf{U}
\end{array}
\qquad\qquad
\begin{array}{ccc}
\sum\limits_{A:\mathsf{U}} \Omega^A & \xrightarrow{\quad\forall\quad} & \Omega \\
{\scriptstyle Pi}\downarrow & & \uparrow{\scriptstyle s} \\
\sum\limits_{A:\mathsf{U}} \mathsf{U}^A & \xrightarrow[\quad\Pi\quad]{} & \mathsf{U}
\end{array}
$$

□

References

1. Awodey, S. (2017) Natural models of homotopy type theory, MSCS 28(2). arXiv:1406.3219
2. Awodey, S. and N. Gambino and S. Hazratpour (2022) Kripke-Joyal semantics for homotopy type theory. arXiv:2110.14576
3. Awodey, S. and C. Newstead (2018) Polynomial pseudomonads and dependent type theory. arXiv:1802.00997
4. Dybjer, P. (1995) Internal Type Theory. Types 1995.
5. Joyal, A. (2017) Notes on clans and tribes. arXiv:1710.10238
6. Newstead, C. (2018) Algebraic Models of Dependent Type Theory, CMU PhD thesis. arXiv:2103.06155
7. Spivak, D. (2022) A summary of categorical structures in Poly. arXiv:2202.00534

Appendix: Natural models of HoTT

Theorem

A category $\mathbb{C}$ with a terminal object 1 admits a natural model of Homotopy type theory if it has a class of maps $\mathcal{D}$ satisfying the following conditions:

- **total:** *every $C \to 1$ is in $\mathcal{D}$,*
- **stable:** *$\mathcal{D}$ is closed under pullbacks along all maps in $\mathbb{C}$,*
- **closed:** *$\mathcal{D}$ is closed under composition and under dependent products along all maps in $\mathcal{D}$,*
- **factorizing:** *every map $f : A \to B$ in $\mathbb{C}$ factors as $f = d \circ a$ with $a \in {}^{\pitchfork}\mathcal{D}$ and $d \in \mathcal{D}$.*

Proof.

Uses the main idea of the Lumsdaine-Warren coherence theorem: a left-adjoint splitting of the fibration of $\mathcal{D}$-maps. □

Appendix: Natural models of HoTT

Examples of categories satisfying the conditions of the theorem:

- Kan complexes with the fibration wfs on sSets.
- Any right-proper Cisinski model category (restricted to the fibrant objects).
- Groupoids, n-Groupoids, ∞-Groupoids.
- Joyal's πh-tribes.
- The syntactic category of contexts of type theory itself.