A model of linear logic using polynomials in Homotopy Type Theory

Elies Harington Samuel Mimram

École Polytechnique

21st of june
MFPS 2024

Overview

- 1988 : Normal functors, power series and λ-calculus - Jean-Yves Girard

Overview

- 1988 : Normal functors, power series and λ-calculus - Jean-Yves Girard
- Normal functors form a 2-category, cartesian closed but not 2-cartesian closed.

Overview

- 1988 : Normal functors, power series and λ-calculus - Jean-Yves Girard
- Normal functors form a 2-category, cartesian closed but not 2-cartesian closed.
- MFPS 2021 : A Cartesian Bicategory of Polynomial Functors in Homotopy Type Theory (Finster, Mimram, Lucas, Seiller).

Overview

- 1988 : Normal functors, power series and λ-calculus - Jean-Yves Girard
- Normal functors form a 2-category, cartesian closed but not 2-cartesian closed.
- MFPS 2021 : A Cartesian Bicategory of Polynomial Functors in Homotopy Type Theory (Finster, Mimram, Lucas, Seiller).
- Our contribution : polynomial functors in types form a model of linear logic.

Overview

- 1988 : Normal functors, power series and λ-calculus - Jean-Yves Girard
- Normal functors form a 2-category, cartesian closed but not 2-cartesian closed.
- MFPS 2021 : A Cartesian Bicategory of Polynomial Functors in Homotopy Type Theory (Finster, Mimram, Lucas, Seiller).
- Our contribution : polynomial functors in types form a model of linear logic.
- Linear logic : "$(A \implies B) \simeq (!A \multimap B)$"

Polynomials in sets

A polynomial in sets is an expression

$$P(X) = \sum_{b \in B} X^{E_b}$$

where B is a set and $(E_b)_{b \in B}$ a family of sets indexed by B.

Polynomials in sets

A polynomial in sets is an expression

$$P(X) = \sum_{b \in B} X^{E_b}$$

where B is a set and $(E_b)_{b \in B}$ a family of sets indexed by B.
No coefficients, but repetitions allowed :

$$3 \times X^{\mathbb{2}} = X^{\mathbb{2}} \sqcup X^{\mathbb{2}} \sqcup X^{\mathbb{2}}$$

Polynomials in sets

$$(B, (E_b)_{b \in B}) \iff (E \to B)$$

so the data of a polynomial P can be encoded as a map of sets

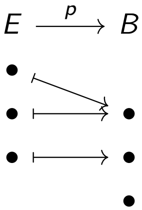
$$E \xrightarrow{p} B$$

- $b \in B$ represents a monomial
- $E_b := p^{-1}(b)$ is the exponent of the monomial b

It induces a **polynomial functor**

$$\mathbf{Set} \to \mathbf{Set}$$
$$X \mapsto \sum_{b \in B} X^{E_b}$$

Example

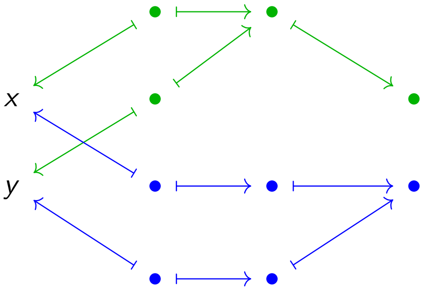


Induced functor:

$$\mathbf{Set} \to \mathbf{Set}$$
$$X \mapsto X^2 \sqcup X \sqcup \mathbb{1}$$

Polynomials with sorts/colors

$$\{x, y\} \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} \{0, 1\}$$



$$\mathbf{Set}^{\{x,y\}} \to \mathbf{Set}^{\{0,1\}}$$

$$(X, Y) \mapsto (XY, X + Y)$$

Composition of polynomials

Generally, a diagram of sets

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

induces a polynomial functor

$$P : \mathbf{Set}^I \to \mathbf{Set}^J$$

Composition of polynomials

Generally, a diagram of sets

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

induces a polynomial functor

$$P : \mathbf{Set}^I \to \mathbf{Set}^J$$

- If $P : \mathbf{Set}^I \to \mathbf{Set}^J$ and $Q : \mathbf{Set}^J \to \mathbf{Set}^K$ are polynomial, so is $Q \circ P$.

Composition of polynomials

Generally, a diagram of sets

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

induces a polynomial functor

$$P : \mathbf{Set}^I \to \mathbf{Set}^J$$

- If $P : \mathbf{Set}^I \to \mathbf{Set}^J$ and $Q : \mathbf{Set}^J \to \mathbf{Set}^K$ are polynomial, so is $Q \circ P$.
- Composition can already be defined at the level of diagrams, but it's only associative/unital up to isomorphism.

Composition of polynomials

Generally, a diagram of sets

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

induces a polynomial functor

$$P : \mathbf{Set}^I \to \mathbf{Set}^J$$

- If $P : \mathbf{Set}^I \to \mathbf{Set}^J$ and $Q : \mathbf{Set}^J \to \mathbf{Set}^K$ are polynomial, so is $Q \circ P$.
- Composition can already be defined at the level of diagrams, but it's only associative/unital up to isomorphism.
- Polynomials in sets thus form a **(2,1)-category**.

Linear polynomials: spans

- A polynomial $P(X) = \sum_{b \in B} X^{E_b}$ is **linear** when E_b is a singleton for all b.

Linear polynomials: spans

- A polynomial $P(X) = \sum_{b \in B} X^{E_b}$ is **linear** when E_b is a singleton for all b.
- In diagram form, this means $p : E \to B$ is an isomorphism.
- Up to isomorphism of polynomials, we can take $p = \mathrm{id}_B$, so linear polynomials correspond to spans

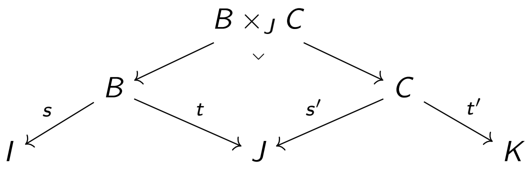
$$I \xleftarrow{s} B \xrightarrow{t} J \qquad \Longleftrightarrow \qquad I \xleftarrow{s} B \xrightarrow{\mathrm{id}_B} B \xrightarrow{t} J$$

Linear polynomials: spans

- A polynomial $P(X) = \sum_{b \in B} X^{E_b}$ is **linear** when E_b is a singleton for all b.
- In diagram form, this means $p : E \to B$ is an isomorphism.
- Up to isomorphism of polynomials, we can take $p = \mathrm{id}_B$, so linear polynomials correspond to spans

$$I \xleftarrow{s} B \xrightarrow{t} J \qquad \Longleftrightarrow \qquad I \xleftarrow{s} B \xrightarrow{\mathrm{id}_B} B \xrightarrow{t} J$$

And they compose via pullbacks



A model of linear logic ?

Spans are linear polynomials. To get a model of LL, we would like

$$(A \implies B) \simeq (!A \multimap B)$$
$$\mathbf{Poly}(I, J) \simeq \mathbf{Span}(!I, J)$$

for a suitable functor $! : \mathbf{Span} \to \mathbf{Span}$.

A model of linear logic ?

Spans are linear polynomials. To get a model of LL, we would like

$$(A \implies B) \simeq (!A \multimap B)$$
$$\mathbf{Poly}(I, J) \simeq \mathbf{Span}(!I, J)$$

for a suitable functor $! : \mathbf{Span} \to \mathbf{Span}$.
A polynomial functor $P : \mathbf{Set} \to \mathbf{Set}$ is determined by $B : \mathbf{Set}$, $E_{(-)} : B \to \mathbf{Set}$, so by a span

$$\mathbf{Set} \xleftarrow[E_{(-)}]{} B \longrightarrow \mathbb{1}$$

- This suggests $!(\mathbb{1}) \simeq \mathbf{Set}$, but there is not set of all sets.
- So we restrict the size of the sets E_b.

Finitary polynomials

A polynomial is **finitary** if the E_b's in its definition are finite sets.
Notation : $\mathbf{Poly}_{\mathrm{Fin}}(I, J)$

Finitary polynomials

A polynomial is **finitary** if the E_b's in its definition are finite sets.
Notation : $\mathbf{Poly}_{\mathrm{Fin}}(I, J)$
Examples:

- $X \mapsto X^3 + X + 1$ is finitary
- $X \mapsto \mathbb{N} \times X$ is finitary
- $(X_k)_{k\in\mathbb{N}} \mapsto ((X_k)^k)_{k\in\mathbb{N}}$ is finitary
- $X \mapsto X^{\mathbb{N}}$ is **not** finitary
- a linear polynomial is always finitary

A model of linear logic ??

A finite set E is characterized by its cardinality $\#E \in \mathbb{N}$. Do we now have

$$\mathbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1}) \simeq \mathbf{Span}(\mathbb{N}, \mathbb{1}) \text{ ?}$$

A model of linear logic ??

A finite set E is characterized by its cardinality $\#E \in \mathbb{N}$. Do we now have

$$\mathbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1}) \simeq \mathbf{Span}(\mathbb{N}, \mathbb{1}) \ ?$$

- $\mathbf{Poly}_{\mathrm{Fin}}$ and **Span** are $(2, 1)$-categories

A model of linear logic ??

A finite set E is characterized by its cardinality $\#E \in \mathbb{N}$. Do we now have

$$\textbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1}) \simeq \textbf{Span}(\mathbb{N}, \mathbb{1}) \ ?$$

- $\textbf{Poly}_{\mathrm{Fin}}$ and **Span** are $(2,1)$-categories
- So $\textbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1})$ and $\textbf{Span}(\mathbb{N}, \mathbb{1})$ are groupoids.

A model of linear logic ??

A finite set E is characterized by its cardinality $\#E \in \mathbb{N}$. Do we now have

$$\mathbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1}) \simeq \mathbf{Span}(\mathbb{N}, \mathbb{1}) \ ?$$

- $\mathbf{Poly}_{\mathrm{Fin}}$ and **Span** are $(2, 1)$-categories
- So $\mathbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1})$ and $\mathbf{Span}(\mathbb{N}, \mathbb{1})$ are groupoids.
- Isomorphism classes are in bijection, but not an equivalence of groupoids :

A model of linear logic ??

A finite set E is characterized by its cardinality $\#E \in \mathbb{N}$. Do we now have

$$\textbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1}) \simeq \textbf{Span}(\mathbb{N}, \mathbb{1}) \ ?$$

- $\textbf{Poly}_{\mathrm{Fin}}$ and **Span** are $(2, 1)$-categories
- So $\textbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1})$ and $\textbf{Span}(\mathbb{N}, \mathbb{1})$ are groupoids.
- Isomorphism classes are in bijection, but not an equivalence of groupoids :
- $P(X) = X \times X$ has a non trivial automorphism $(a, b) \mapsto (b, a)$

A model of linear logic ??

A finite set E is characterized by its cardinality $\#E \in \mathbb{N}$. Do we now have

$$\mathbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1}) \simeq \mathbf{Span}(\mathbb{N}, \mathbb{1}) \text{ ?}$$

- $\mathbf{Poly}_{\mathrm{Fin}}$ and $\mathbf{Span}$ are $(2, 1)$-categories
- So $\mathbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1})$ and $\mathbf{Span}(\mathbb{N}, \mathbb{1})$ are groupoids.
- Isomorphism classes are in bijection, but not an equivalence of groupoids :
- $P(X) = X \times X$ has a non trivial automorphism $(a, b) \mapsto (b, a)$
- the corresponding span $\mathbb{N} \xleftarrow{2 \mapsfrom *} \mathbb{1} \longrightarrow \mathbb{1}$ doesn't have any

A model of linear logic ??

A finite set E is characterized by its cardinality $\#E \in \mathbb{N}$. Do we now have

$$\textbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1}) \simeq \textbf{Span}(\mathbb{N}, \mathbb{1}) \ ?$$

- $\textbf{Poly}_{\mathrm{Fin}}$ and $\textbf{Span}$ are $(2, 1)$-categories
- So $\textbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1})$ and $\textbf{Span}(\mathbb{N}, \mathbb{1})$ are groupoids.
- Isomorphism classes are in bijection, but not an equivalence of groupoids :
- $P(X) = X \times X$ has a non trivial automorphism $(a, b) \mapsto (b, a)$
- the corresponding span $\mathbb{N} \overset{2 \leftarrow\!\shortmid *}{\longleftarrow} \mathbb{1} \longrightarrow \mathbb{1}$ doesn't have any
- Need to replace $\mathbb{N}$ by the groupoid Fin of finite sets and bijections.

Polynomials in groupoids

- Already observed by Finster et. al. that groupoids are necessary for a cartesian closure of polynomials.
  Here we reach the same observation.

Polynomials in groupoids

- Already observed by Finster et. al. that groupoids are necessary for a cartesian closure of polynomials.
  Here we reach the same observation.
- However not easy to work with groupoids : not even a LCC category. . .

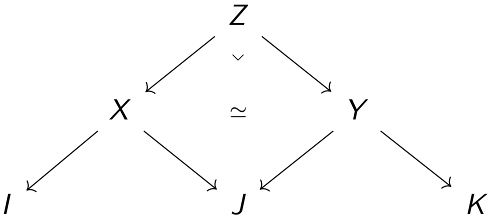
Polynomials in groupoids

- Already observed by Finster et. al. that groupoids are necessary for a cartesian closure of polynomials.
  Here we reach the same observation.
- However not easy to work with groupoids : not even a LCC category...
- but it is LCC **up to homotopy** !

Polynomials in groupoids

- Already observed by Finster et. al. that groupoids are necessary for a cartesian closure of polynomials.
  Here we reach the same observation.
- However not easy to work with groupoids : not even a LCC category...
- but it is LCC **up to homotopy** !

Working homotopically is essential : composition of spans of groupoids needs to be done by **homotopy pullback**.



Homotopy Type Theory

Working homotopically can be cumbersome in set theory, hence we work in **Homotopy Type Theory** :

- An extension of Martin Löf type theory,

Homotopy Type Theory

Working homotopically can be cumbersome in set theory, hence we work in **Homotopy Type Theory** :

- An extension of Martin Löf type theory,
- types are thought of as **spaces**, in the sense of **homotopy theory**

Homotopy Type Theory

Working homotopically can be cumbersome in set theory, hence we work in **Homotopy Type Theory** :

- An extension of Martin Löf type theory,
- types are thought of as **spaces**, in the sense of **homotopy theory**
- e.g. sets are types, groupoids are types, higher groupoids are types. . .

Working homotopically can be cumbersome in set theory, hence we work in **Homotopy Type Theory** :

- An extension of Martin Löf type theory,
- types are thought of as **spaces**, in the sense of **homotopy theory**
- e.g. sets are types, groupoids are types, higher groupoids are types. . .
- all maps between types are automatically functorial

Homotopy Type Theory

Working homotopically can be cumbersome in set theory, hence we work in **Homotopy Type Theory** :

- An extension of Martin Löf type theory,
- types are thought of as **spaces**, in the sense of **homotopy theory**
- e.g. sets are types, groupoids are types, higher groupoids are types. . .
- all maps between types are automatically functorial
- all constructions inside HoTT are automatically **homotopy invariant** :

Homotopy Type Theory

Working homotopically can be cumbersome in set theory, hence we work in **Homotopy Type Theory** :

- An extension of Martin Löf type theory,
- types are thought of as **spaces**, in the sense of **homotopy theory**
- e.g. sets are types, groupoids are types, higher groupoids are types...
- all maps between types are automatically functorial
- all constructions inside HoTT are automatically **homotopy invariant** :
  - quotients are homotopy quotients : they add homotopies instead of equating elements

Homotopy Type Theory

Working homotopically can be cumbersome in set theory, hence we work in **Homotopy Type Theory** :

- An extension of Martin Löf type theory,
- types are thought of as **spaces**, in the sense of **homotopy theory**
- e.g. sets are types, groupoids are types, higher groupoids are types...
- all maps between types are automatically functorial
- all constructions inside HoTT are automatically **homotopy invariant** :
  - quotients are homotopy quotients : they add homotopies instead of equating elements
  - Σ-types are Grothendieck constructions

Working homotopically can be cumbersome in set theory, hence we work in **Homotopy Type Theory** :

- An extension of Martin Löf type theory,
- types are thought of as **spaces**, in the sense of **homotopy theory**
- e.g. sets are types, groupoids are types, higher groupoids are types. . .
- all maps between types are automatically functorial
- all constructions inside HoTT are automatically **homotopy invariant** :
  - quotients are homotopy quotients : they add homotopies instead of equating elements
  - Σ-types are Grothendieck constructions
  - fibers are homotopy fibers, pullbacks are homotopy pullbacks. . .

Spans and polynomials in HoTT

- Polynomials and spans in HoTT : same as before, but replace "set" with "type".
- A polynomial is **finitary** if the E_b's are finite **sets** (seen as types).
- Write Fin for the type of finite sets and bijections.

$$\mathbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1}) \simeq \mathbf{Span}(\mathrm{Fin}, \mathbb{1})$$

so $!\mathbb{1} \simeq \mathrm{Fin}$.

Spans and polynomials in HoTT

- Polynomials and spans in HoTT : same as before, but replace "set" with "type".
- A polynomial is **finitary** if the E_b's are finite **sets** (seen as types).
- Write Fin for the type of finite sets and bijections.

$$\mathbf{Poly}_{\mathrm{Fin}}(\mathbb{1}, \mathbb{1}) \simeq \mathbf{Span}(\mathrm{Fin}, \mathbb{1})$$

so $!\mathbb{1} \simeq \mathrm{Fin}$.
More generally, define $!I :=$ finite sets colored by I, i.e.

$$!I \simeq \sum_{E:\mathrm{Fin}} (E \to I)$$

Theorem

$$\mathbf{Poly}_{\mathrm{Fin}}(I, J) \simeq \mathbf{Span}(!I, J)$$

Idea of proof

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

$$\begin{aligned}
\mathbf{Poly}_{\mathrm{Fin}}(I, J) &= \sum_{E:\mathcal{U}} \sum_{B:\mathcal{U}} (E \to I) \times (E \to_{\mathrm{Fin}} B) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{E:\mathcal{U}} (E \to_{\mathrm{Fin}} B) \times (E \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{F:B\to\mathrm{Fin}} ((\Sigma_{b:B} F(b)) \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(B \to \sum_{F:\mathrm{Fin}} (F \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} (B \to !I) \times (B \to J) \\
&= \mathbf{Span}(!I, J)
\end{aligned}$$

Idea of proof

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

$$\mathbf{Poly}_{\mathrm{Fin}}(I, J) = \sum_{E:\mathcal{U}} \sum_{B:\mathcal{U}} (E \to I) \times (E \to_{\mathrm{Fin}} B) \times (B \to J)$$

$$\simeq \sum_{B:\mathcal{U}} \left(\sum_{E:\mathcal{U}} (E \to_{\mathrm{Fin}} B) \times (E \to I) \right) \times (B \to J)$$

$$\simeq \sum_{B:\mathcal{U}} \left(\sum_{F:B \to \mathrm{Fin}} ((\Sigma_{b:B} F(b)) \to I) \right) \times (B \to J)$$

$$\simeq \sum_{B:\mathcal{U}} \left(B \to \sum_{F:\mathrm{Fin}} (F \to I) \right) \times (B \to J)$$

$$\simeq \sum_{B:\mathcal{U}} (B \to !I) \times (B \to J)$$

$$= \mathbf{Span}(!I, J)$$

Idea of proof

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

$$\begin{aligned}
\mathbf{Poly}_{\mathrm{Fin}}(I,J) &= \sum_{E:\mathcal{U}} \sum_{B:\mathcal{U}} (E \to I) \times (E \to_{\mathrm{Fin}} B) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{E:\mathcal{U}} (E \to_{\mathrm{Fin}} B) \times (E \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{F:B \to \mathrm{Fin}} ((\Sigma_{b:B} F(b)) \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(B \to \sum_{F:\mathrm{Fin}} (F \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} (B \to !I) \times (B \to J) \\
&= \mathbf{Span}(!I, J)
\end{aligned}$$

Idea of proof

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

$$\begin{aligned}
\mathbf{Poly}_{\mathrm{Fin}}(I, J) &= \sum_{E:\mathcal{U}} \sum_{B:\mathcal{U}} (E \to I) \times (E \to_{\mathrm{Fin}} B) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{E:\mathcal{U}} (E \to_{\mathrm{Fin}} B) \times (E \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{F:B\to\mathrm{Fin}} ((\Sigma_{b:B} F(b)) \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(B \to \sum_{F:\mathrm{Fin}} (F \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} (B \to !I) \times (B \to J) \\
&= \mathbf{Span}(!I, J)
\end{aligned}$$

Idea of proof

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

$$\begin{aligned}
\mathbf{Poly}_{\mathrm{Fin}}(I,J) &= \sum_{E:\mathcal{U}} \sum_{B:\mathcal{U}} (E \to I) \times (E \to_{\mathrm{Fin}} B) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{E:\mathcal{U}} (E \to_{\mathrm{Fin}} B) \times (E \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{F:B\to\mathrm{Fin}} ((\Sigma_{b:B} F(b)) \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(B \to \sum_{F:\mathrm{Fin}} (F \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} (B \to !I) \times (B \to J) \\
&= \mathbf{Span}(!I, J)
\end{aligned}$$

Idea of proof

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

$$\begin{aligned}
\mathbf{Poly}_{\mathrm{Fin}}(I, J) &= \sum_{E:\mathcal{U}} \sum_{B:\mathcal{U}} (E \to I) \times (E \to_{\mathrm{Fin}} B) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \Big(\sum_{E:\mathcal{U}} (E \to_{\mathrm{Fin}} B) \times (E \to I) \Big) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \Big(\sum_{F:B \to \mathrm{Fin}} ((\Sigma_{b:B} F(b)) \to I) \Big) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \Big(B \to \sum_{F:\mathrm{Fin}} (F \to I) \Big) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} (B \to !I) \times (B \to J) \\
&= \mathbf{Span}(!I, J)
\end{aligned}$$

Idea of proof

$$I \xleftarrow{s} E \xrightarrow{p} B \xrightarrow{t} J$$

$$\begin{aligned}
\mathbf{Poly}_{\mathrm{Fin}}(I, J) &= \sum_{E:\mathcal{U}} \sum_{B:\mathcal{U}} (E \to I) \times (E \to_{\mathrm{Fin}} B) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{E:\mathcal{U}} (E \to_{\mathrm{Fin}} B) \times (E \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(\sum_{F:B \to \mathrm{Fin}} ((\Sigma_{b:B} F(b)) \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} \left(B \to \sum_{F:\mathrm{Fin}} (F \to I) \right) \times (B \to J) \\
&\simeq \sum_{B:\mathcal{U}} (B \to !I) \times (B \to J) \\
&= \mathbf{Span}(!I, J)
\end{aligned}$$

A parenthesis : higher categorical coherences

- Switching from sets to groupoids makes **Poly** and **Span** into 3-categories.

A parenthesis : higher categorical coherences

- Switching from sets to groupoids makes **Poly** and **Span** into 3-categories.
- Going to arbitrary types, we get an ∞-category : associativity and unitality up to isomorphisms, themselves satisfying coherence laws, etc.

A parenthesis : higher categorical coherences

- Switching from sets to groupoids makes **Poly** and **Span** into 3-categories.
- Going to arbitrary types, we get an ∞-category : associativity and unitality up to isomorphisms, themselves satisfying coherence laws, etc.
- We cannot state or prove those infinite **homotopy coherence laws** in HoTT, so we work with **wild categories**.

A parenthesis : higher categorical coherences

- Switching from sets to groupoids makes **Poly** and **Span** into 3-categories.
- Going to arbitrary types, we get an ∞-category : associativity and unitality up to isomorphisms, themselves satisfying coherence laws, etc.
- We cannot state or prove those infinite **homotopy coherence laws** in HoTT, so we work with **wild categories**.
- Wild categories have the standard definition of categories, but with sets replaced by types.

A parenthesis : higher categorical coherences

- Switching from sets to groupoids makes **Poly** and **Span** into 3-categories.
- Going to arbitrary types, we get an ∞-category : associativity and unitality up to isomorphisms, themselves satisfying coherence laws, etc.
- We cannot state or prove those infinite **homotopy coherence laws** in HoTT, so we work with **wild categories**.
- Wild categories have the standard definition of categories, but with sets replaced by types.
- No pentagon or triangle isomorphisms required of the associators and unitors.

A parenthesis : higher categorical coherences

- Switching from sets to groupoids makes **Poly** and **Span** into 3-categories.
- Going to arbitrary types, we get an ∞-category : associativity and unitality up to isomorphisms, themselves satisfying coherence laws, etc.
- We cannot state or prove those infinite **homotopy coherence laws** in HoTT, so we work with **wild categories**.
- Wild categories have the standard definition of categories, but with sets replaced by types.
- No pentagon or triangle isomorphisms required of the associators and unitors.

Remark

Not all coherences can be stated in HoTT, but some can be proven meta-theoretically. For instance, we can prove a wild category has cartesian products, and know meta-theoretically that the induced monoidal structure is homotopy coherent.

A full model of classical linear logic

Proposition

! defines a comonad on the wild category of spans of types.

A full model of classical linear logic

Proposition

! defines a comonad on the wild category of spans of types.

Theorem

The Kleisli category $\mathbf{Span}_!$ is equivalent to $\mathbf{Poly}_{\mathrm{Fin}}$.

A full model of classical linear logic

Proposition

! defines a comonad on the wild category of spans of types.

Theorem

The Kleisli category $\mathbf{Span}_!$ is equivalent to $\mathbf{Poly}_{\mathrm{Fin}}$.

Theorem

$(\mathbf{Span}, !)$ has a compatible symmetric monoidal structure, making it a **Seely category** (model of intuitionistic linear logic).
It is moreover compact closed, hence $\star$-autonomous, i.e. a model of full classical linear logic.

A full model of classical linear logic

Proposition

! defines a comonad on the wild category of spans of types.

Theorem

The Kleisli category $\mathbf{Span}_!$ is equivalent to $\mathbf{Poly}_{\mathrm{Fin}}$.

Theorem

$(\mathbf{Span}, !)$ has a compatible symmetric monoidal structure, making it a **Seely category** (model of intuitionistic linear logic).
It is moreover compact closed, hence $\star$-autonomous, i.e. a model of full classical linear logic.

Corollary

Since $(\mathbf{Span}, !)$ is monoidal closed, $\mathbf{Poly}_{\mathrm{Fin}}$ is cartesian closed.

Variations on size

We had to restrict to finitary polynomials.

Actually, any suitable notion of “small” gives **smallary** polynomials, with results analogous to everything before.

Given a universe of “big types” $\mathcal{U}$ and $\mathcal{V} : \mathcal{U}$ a universe of “small types” closed under suitable constructions :

- a polynomial is $\mathcal{V}$-ary if the E_b's are equivalent to types in $\mathcal{V}$
- define $!_{\mathcal{V}} I := \sum_{E:\mathcal{V}} (E \to I)$

Variations on size

We had to restrict to finitary polynomials.

Actually, any suitable notion of "small" gives **smallary** polynomials, with results analogous to everything before.

Given a universe of "big types" $\mathcal{U}$ and $\mathcal{V} : \mathcal{U}$ a universe of "small types" closed under suitable constructions :

- a polynomial is $\mathcal{V}$-ary if the E_b's are equivalent to types in $\mathcal{V}$
- define $!_{\mathcal{V}} I := \sum_{E:\mathcal{V}} (E \to I)$

Theorem

$$\mathbf{Poly}_{\mathcal{V}}(I, J) \simeq \mathbf{Span}(!_{\mathcal{V}} I, J)$$

and we get a model of classical linear logic.

- $\mathcal{V} = \mathrm{Fin}$, $\mathcal{U} =$ groupoids
- $\mathcal{V} = \mathbf{Set}$, $\mathcal{U} =$ large groupoids
- $\mathcal{V} =$ groupoids, $\mathcal{U} =$ all types

Examples of higher polynomials - 1

$$\mathrm{Fin}_* \xrightarrow{p} \mathrm{Fin}$$

$$(E, e) \longmapsto E$$

Examples of higher polynomials - 1

$$\mathrm{Fin}_* \xrightarrow{p} \mathrm{Fin}$$

$$(E, e) \longmapsto E$$

- Given $E : \mathrm{Fin}$, $p^{-1}(E) \simeq E$

Examples of higher polynomials - 1

$$\mathrm{Fin}_* \xrightarrow{p} \mathrm{Fin}$$

$$(E, e) \longmapsto E$$

- Given $E : \mathrm{Fin}$, $p^{-1}(E) \simeq E$
- Induced polynomial : $F(X) = \sum_{E:\mathrm{Fin}} X^E$
- This exactly the definition $!_{\mathrm{Fin}}$: the comonad is itself a polynomial functor.

Examples of higher polynomials - 2

$$\mathbb{N} \xrightarrow{p} \sum_{n:\mathbb{N}} B(\mathbb{Z}/n\mathbb{Z})$$

$$n \longmapsto (n, \star)$$

Examples of higher polynomials - 2

$$\mathbb{N} \xrightarrow{p} \sum_{n:\mathbb{N}} B(\mathbb{Z}/n\mathbb{Z})$$

$$n \longmapsto (n, \star)$$

- $B(\mathbb{Z}/n\mathbb{Z})$ is the groupoid with one point and $\mathbb{Z}/n\mathbb{Z}$ as automorphisms

Examples of higher polynomials - 2

$$\mathbb{N} \xrightarrow{p} \sum_{n:\mathbb{N}} B(\mathbb{Z}/n\mathbb{Z})$$

$$n \longmapsto (n, \star)$$

- $B(\mathbb{Z}/n\mathbb{Z})$ is the groupoid with one point and $\mathbb{Z}/n\mathbb{Z}$ as automorphisms
- $p^{-1}((n, \star)) \simeq \mathbb{Z}/n\mathbb{Z}$

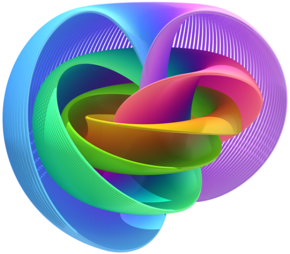
Examples of higher polynomials - 2

$$\mathbb{N} \xrightarrow{p} \sum_{n:\mathbb{N}} B(\mathbb{Z}/n\mathbb{Z})$$

$$n \longmapsto (n, \star)$$

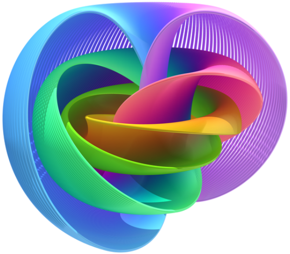
- $B(\mathbb{Z}/n\mathbb{Z})$ is the groupoid with one point and $\mathbb{Z}/n\mathbb{Z}$ as automorphisms
- $p^{-1}((n,\star)) \simeq \mathbb{Z}/n\mathbb{Z}$
- Induced polynomial : $F(X) = \sum_{n:\mathbb{N}} X^n /\!\!/ (\mathbb{Z}/n\mathbb{Z})$
- The type of cyclic lists over X
- Generally, summing over groupoids amounts to quotienting the summand

Examples of higher polynomials - The Hopf Fibration



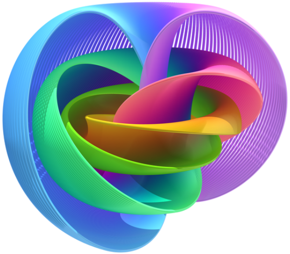
$$S^3 \xrightarrow{H} S^2$$

Examples of higher polynomials - The Hopf Fibration



$$S^3 \xrightarrow{H} S^2$$

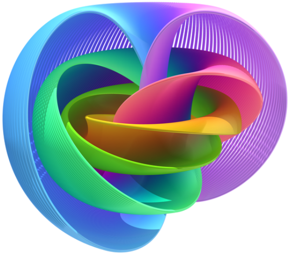
- $\left\| H^{-1}(x) \simeq S^1 \right\|_{-1}$ but $(\sum_{x:S^2} H^{-1}(x)) \simeq S^3 \not\simeq S^2 \times S^1$



$$S^3 \xrightarrow{H} S^2$$

- $\| H^{-1}(x) \simeq S^1 \|_{-1}$ but $(\sum_{x:S^2} H^{-1}(x)) \simeq S^3 \not\simeq S^2 \times S^1$
- $F(X) = \sum_{x:S^2} X^{H^{-1}(x)}$.

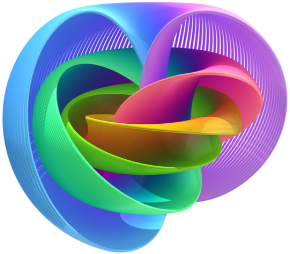
Examples of higher polynomials - The Hopf Fibration



$$S^3 \xrightarrow{H} S^2$$

- $\left\| H^{-1}(x) \simeq S^1 \right\|_{-1}$ but $(\sum_{x:S^2} H^{-1}(x)) \simeq S^3 \not\simeq S^2 \times S^1$
- $F(X) = \sum_{x:S^2} X^{H^{-1}(x)}$.
- This locally looks like $S^2 \times X^{S^1}$, but in a globally twisted way.

Examples of higher polynomials - The Hopf Fibration



$$S^3 \xrightarrow{H} S^2$$

- $\| H^{-1}(x) \simeq S^1 \|_{-1}$ but $(\sum_{x:S^2} H^{-1}(x)) \simeq S^3 \not\simeq S^2 \times S^1$
- $F(X) = \sum_{x:S^2} X^{H^{-1}(x)}$.
- This locally looks like $S^2 \times X^{S^1}$, but in a globally twisted way.
- If you have any idea what this represents, please reach out !

Conclusion/future work

We arranged the usual notions of spans and polynomials into a model of linear logic, using ideas from homotopy type theory. What next?

Conclusion/future work

We arranged the usual notions of spans and polynomials into a model of linear logic, using ideas from homotopy type theory. What next?

- Differential structure?

Conclusion/future work

We arranged the usual notions of spans and polynomials into a model of linear logic, using ideas from homotopy type theory. What next?

- Differential structure?
- Exploring other potential homotopically-flavoured models of linear logic: spectra? stable ∞-categories?

Conclusion/future work

We arranged the usual notions of spans and polynomials into a model of linear logic, using ideas from homotopy type theory. What next?

- Differential structure?
- Exploring other potential homotopically-flavoured models of linear logic: spectra? stable ∞-categories?
- Comparison with other span-based models of linear logic by Mellies, Clairambault, Forest

Conclusion/future work

We arranged the usual notions of spans and polynomials into a model of linear logic, using ideas from homotopy type theory. What next?

- Differential structure?
- Exploring other potential homotopically-flavoured models of linear logic: spectra? stable ∞-categories?
- Comparison with other span-based models of linear logic by Mellies, Clairambault, Forest
- Comparison with generalized species of structure (Fiore, Gambino, Galal, Hyland, Paquet, Winskel)