Operads as double functors

Claudio Pisani

Virtual Double Categories Workshop
December 2, 2022

Table of Contents

Motivations

Aim

A more natural approach to colored operads
(symmetric multicategories).

Achievement

The non-skeletal approach to operads
seems in fact more natural in many respects.

Technical tools

It this approach, double categories play a pivotal role.

Byproduct

This is how I learned to love double categories.

Motivations

The operad of sets

- Objects are sets.
- Arrows $f : X_1; \cdots ; X_n \to Y$
  are maps which take a list of elements
  $x_1; \cdots ; x_n$ (with $x_i \in X_i$)
  and give an element $y \in Y$.

Motivations

The operad of sets

- Objects are sets.
- Arrows $f : X_1; \cdots ; X_n \to Y$
  are maps which take a list of elements
  $x_1; \cdots ; x_n$ (with $x_i \in X_i$)
  and give an element $y \in Y$.

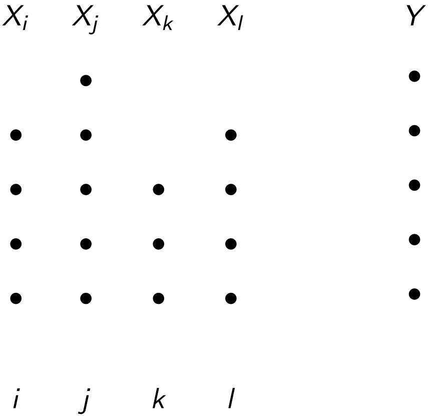
But order doesn't really matter...

Motivations

More natually

- Objects are sets.
- Arrows $f : (X_i)_{i \in A} \to Y$
  are maps which take a family of elements
  $(x_i)_{i \in A}$ (with $x_i \in X_i$)
  and give an element $y \in Y$.

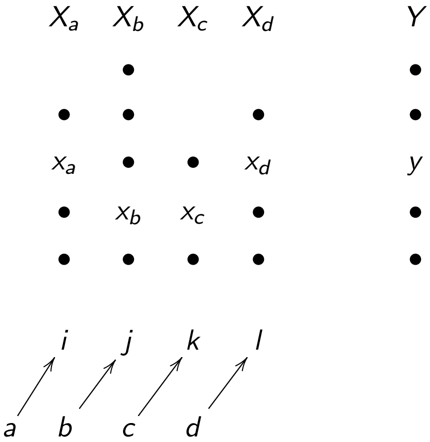
Motivations



Motivations

X_i	X_j	X_k	X_l	Y
	•			•
•	•		•	•
x_i	•	•	x_l	y
•	x_j	x_k	•	•
•	•	•	•	•
i	j	k	l	

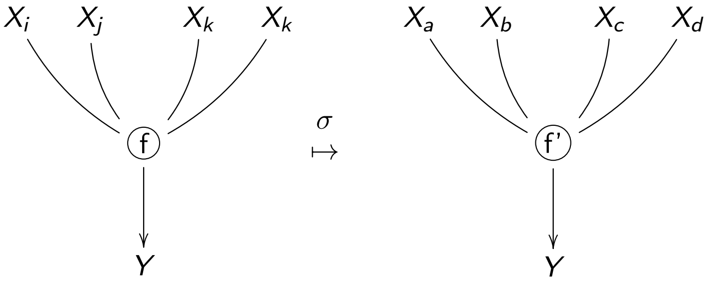
Motivations



Motivations

Reindexing of arrows

A bijection $\sigma : B \to A$ gives a reindexing,
taking any arrow whose domain is indexed by A
to an arrow whose domain is indexed by B.

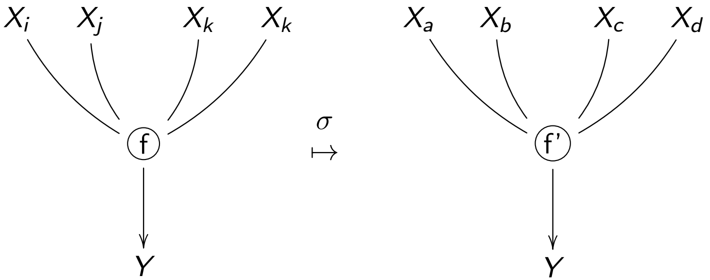


Motivations

Idea

The maps f and f' are the same, up to the indexing of domains.

But indexing is necessary in order to composing arrows.



Motivations

If, instead of single arrows, we consider families of arrows, we get a category with an underlying functor to Set_f.

$$(X_i)_{i\in A} \xrightarrow{(f_j)_{j\in B}} (Y_j)_{j\in B} \xrightarrow{(g_k)_{k\in C}} (Z_k)_{k\in C}, \qquad (X_i)_{i\in A} \xrightarrow[(h_k)_{k\in C}]{} (Z_k)_{k\in C}$$

$$A \xrightarrow{\overline{f}} B \xrightarrow{\overline{g}} C, \qquad A \xrightarrow[\overline{h}]{} C$$

Motivations

Reindexing of families of arrows
are the cells double category.

$$
\begin{array}{ccc}
(X'_j)_{j\in B} & \xleftarrow{\ \sigma^*\ } & (X_i)_{i\in A} \\
\Big\downarrow{\scriptstyle f'} & & \Big\downarrow{\scriptstyle f} \\
(Y'_s)_{s\in D} & \xleftarrow{\ \rho^*\ } & (Y_t)_{t\in C}
\end{array}
$$

Motivations

Main idea

To properly understand operads,
we need a framework allowing to express symmetry of arrows
and yet retaining the possibility of composing them.

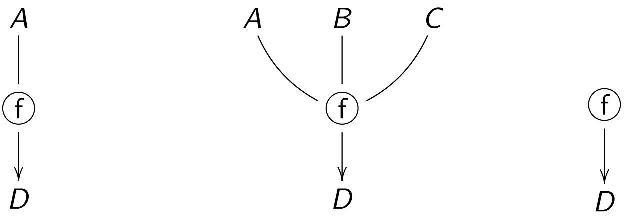
Double categories provide this framework.

Table of Contents

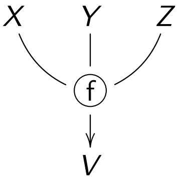
Operads (arity)

Arrows of any arity

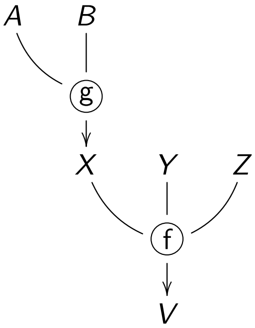
A unary arrow, a ternary arrow and a nullary arrow.



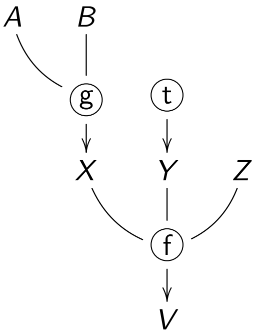
Operads (composition)



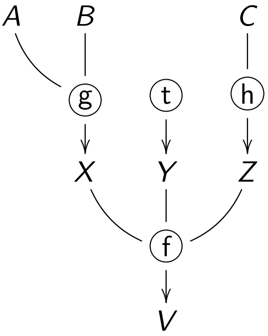
Operads (composition)



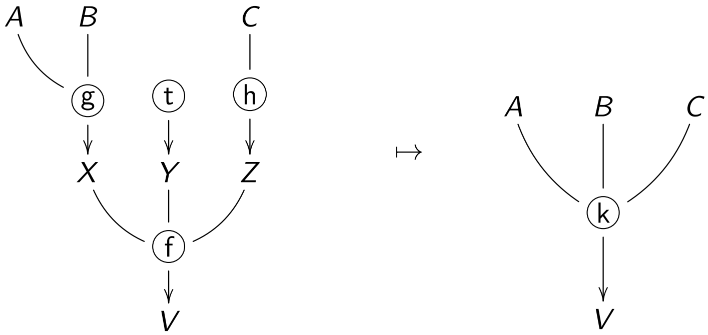
Operads (composition)



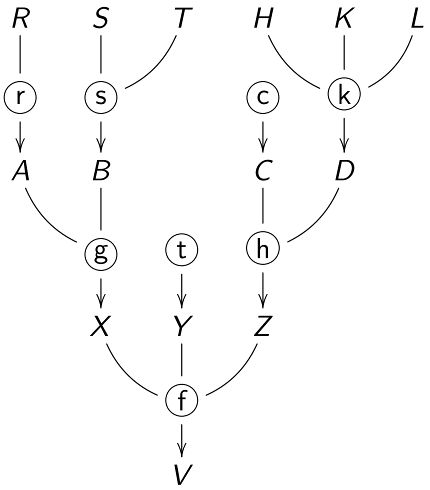
Operads (composition)



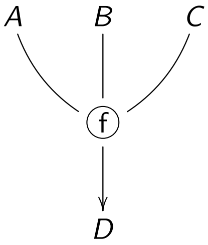
Operads (composition)



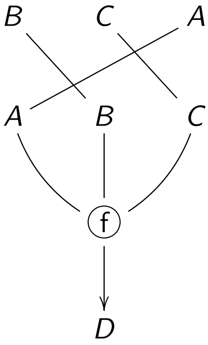
Operads (associativity)



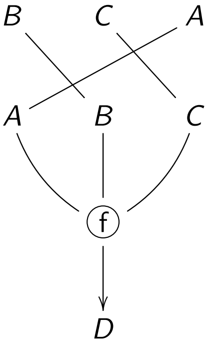
Operads (symmetry)



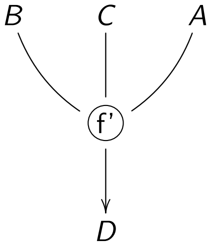
Operads (symmetry)



Operads (symmetry)

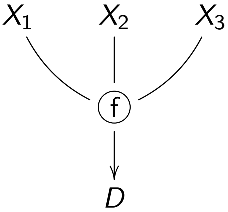


$\mapsto$



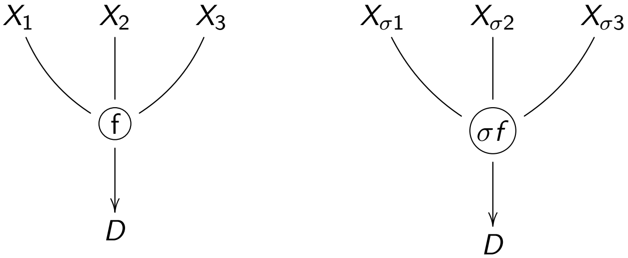
Operads: Classical approach

The domain of an arrow is a list $X : \mathrm{n} \to \mathcal{O}_0$ of objects in $\mathcal{O}$



Operads: Classical approach

Arrows can be transported along permutations σ
of the indexing set $\mathrm{n} = \{1, \cdots, \mathrm{n}\}$



Operads: Classical approach

Axioms

- Composition and associativity.
- Permutations act on arrows.
- The action is compatible with composition.

Operads: Classical approach

Axioms

- Composition and associativity.
- Permutations act on arrows.
- The action is compatible with composition.

When made explicit,
these conditions assume a rather unwieldy form
(involving for instance block permutations)
showing drawbacks of the skeletal choice for indexing.

Operads: Examples

Monoidal operads

Any symmetric monoidal category gives an operad $\mathcal{O}$,
whose arrows $f : X_1; \cdots ; X_n \to Y$
are arrows $f : X_1 \otimes \cdots \otimes X_n \to Y$.

Operads: Examples

Monoidal operads

Any symmetric monoidal category gives an operad $\mathcal{O}$,
whose arrows $f : X_1; \cdots; X_n \to Y$
are arrows $f : X_1 \otimes \cdots \otimes X_n \to Y$.

Can be restricted to an operad $\mathcal{O}'$ for any subset $\mathcal{O}'_0 \subset \mathcal{O}_0$.

Operads: Examples

Monoidal operads

Any symmetric monoidal category gives an operad $\mathcal{O}$, whose arrows $f : X_1; \cdots ; X_n \to Y$ are arrows $f : X_1 \otimes \cdots \otimes X_n \to Y$.

Can be restricted to an operad $\mathcal{O}'$ for any subset $\mathcal{O}'_0 \subset \mathcal{O}_0$.

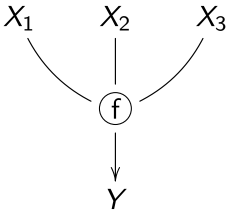
In particular, one can consider a cartesian monoidal category.

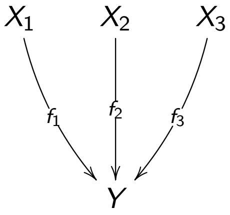
Starting with $(\mathrm{Set}, \times, 1)$ we get the operad of sets.

Operads: Examples

Starting with a cocartesian monoidal category $(\mathcal{C}, +, 0)$
we get the sequential operad $\mathcal{C}_{\blacktriangleright}$ whose maps
are sequences of concurrent arrows in $\mathcal{C}$ (discrete cocones).

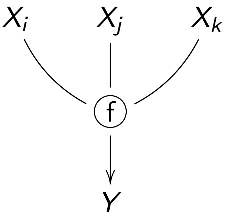
One can consider $\mathcal{C}_{\blacktriangleright}$ for any category $\mathcal{C}$.





Operads: Examples

This example again suggests a more natural notion of operad, where the domain of an arrow is a family of objects
indexed by an arbitrary finite set
(rather than by a set in a skeleton N of Set_f).



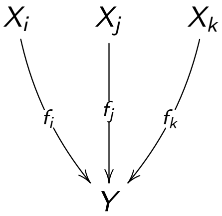
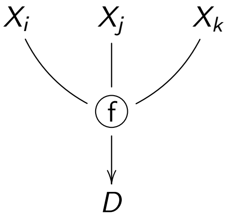


Table of Contents

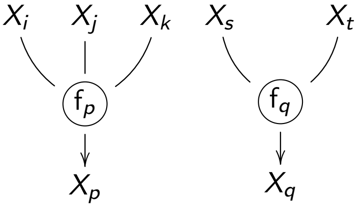
Operads: non-skeletal approach

The domain of an arrow is an arbitrary family
$X : A \to \mathcal{O}_0$ of objects



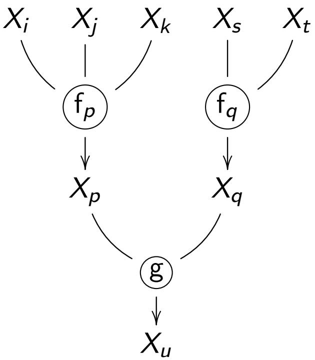
Operads: non-skeletal approach

Taking in account composition,
we need to consider families of arrows.



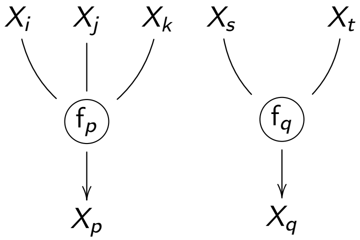
Operads: non-skeletal approach

Taking in account composition,
we need to consider families of arrows.



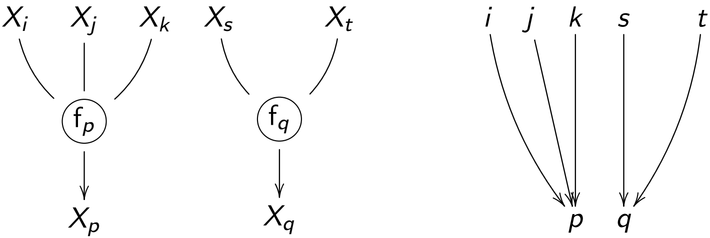
Operads: non-skeletal approach

Any family of arrows has an underlying mapping



Operads: non-skeletal approach

Any family of arrows has an underlying mapping



Operads: non-skeletal approach

Question

So, what do we get by considering families of arrows
in a non-skeletal operad $\mathcal{O}$?

Answer

They form a category $\mathcal{D}_{\mathcal{O}}$ over finite sets:
the functor $d : \mathcal{D}_{\mathcal{O}} \to \mathsf{Set}_f$ keeps track
of the indexing of objects and maps.

Operads: non-skeletal approach

Question

So, what do we get by considering families of arrows in a non-skeletal operad $\mathcal{O}$?

Answer

They form a category $\mathcal{D}_{\mathcal{O}}$ over finite sets: the functor $d : \mathcal{D}_{\mathcal{O}} \to \mathsf{Set}_f$ keeps track of the indexing of objects and maps.

The category $\mathcal{D}_{\mathcal{O}}$, in its skeletal form, appears in the literature under several names, such as "operator" or "envelope" category of $\mathcal{O}$, or the free PROP generated by $\mathcal{O}$.

Operads: non-skeletal approach

Question

What further structure is inherited by $\mathcal{D}_{\mathcal{O}}$
from the operad structure $\mathcal{O}$?

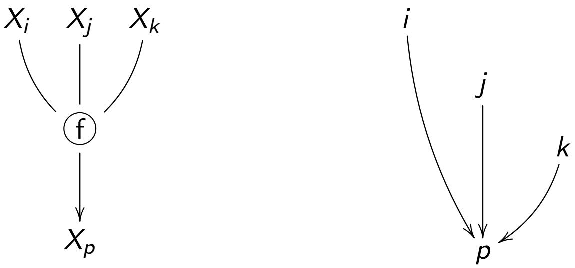
Answer

Its maps (families of arrows in $\mathcal{O}$)
can be reindexed along pullbacks in Set_f.

Reindexing along pullbacks

For instance, we can reindex a single arrow or a family of arrows along pullbacks whose horizontal sides are isomorphisms.

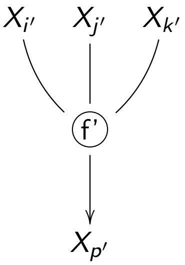
We so obtain the same arrow (up to indexing).

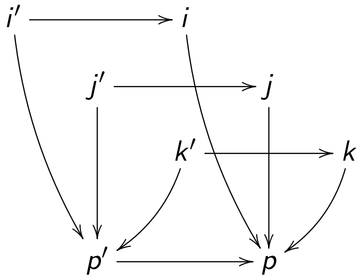


Reindexing along pullbacks

For instance, we can reindex a single arrow or a family of arrows along pullbacks whose horizontal sides are isomorphisms.

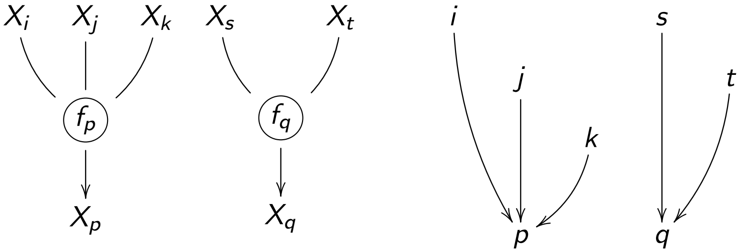
We so obtain the same arrow (up to indexing).





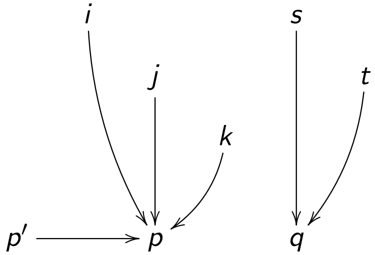
Reindexing along pullbacks

Or we can reindex along injective mappings
to pick up just some arrows of the family.



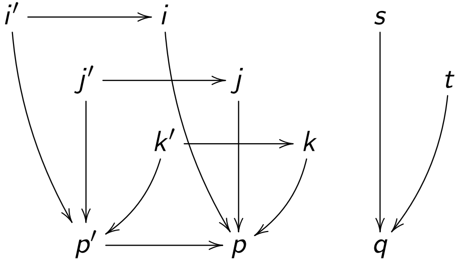
Reindexing along pullbacks

Or we can reindex along injective mappings
to pick up just some arrows of the family.

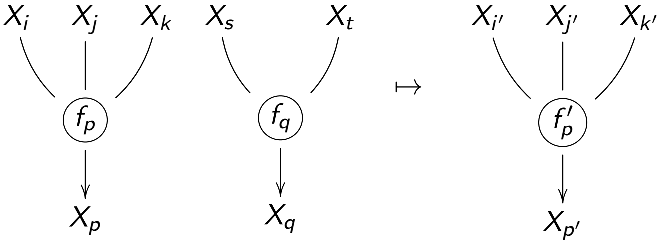


Reindexing along pullbacks

Or we can reindex along injective mappings
to pick up just some arrows of the family.



Reindexing along pullbacks



Reindexing along pullbacks

Or we can reindex along more general mappings
to obtain copies of some of the arrows in a family.

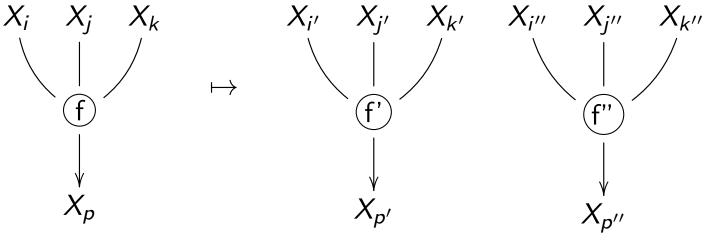


Table of Contents

Question

So, what do we get by considering families of arrows in a non-skeletal operad $\mathcal{O}$?

Answer

Question

So, what do we get by considering families of arrows in a non-skeletal operad $\mathcal{O}$?

Answer

- They form a category over finite sets $d : \mathcal{D}_{\mathcal{O}} \to \mathrm{Set}_f$.

Question

So, what do we get by considering families of arrows in a non-skeletal operad $\mathcal{O}$?

Answer

- They form a category over finite sets $d : \mathcal{D}_{\mathcal{O}} \to \mathsf{Set}_f$.
- For any pullback in Set_f there is a reindexing of objects and of maps over it.

Question

So, what do we get by considering families of arrows in a non-skeletal operad $\mathcal{O}$?

Answer

- They form a category over finite sets $d : \mathcal{D}_{\mathcal{O}} \to \mathsf{Set}_f$.
- For any pullback in Set_f there is a reindexing of objects and of maps over it.
- The reindexing is compatible with composition.

Here they come double categories

For any pullback in Set_f there is a reindexing over it.

Here they come double categories

For any pullback in Set_f there is a reindexing over it.

$$
\begin{array}{c}
X \\
\downarrow{\scriptstyle f} \\
Y
\end{array}
\qquad\qquad
\begin{array}{ccc}
A' & \xrightarrow{s} & A \\
{\scriptstyle k}\downarrow & \text{pb} & \downarrow{\scriptstyle df} \\
B' & \xrightarrow{t} & B
\end{array}
$$

Here they come double categories

For any pullback in Set_f there is a reindexing over it.

s^*X and t^*Y are the reindexing of X and Y along s and t.

Here they come double categories

For any pullback in Set_f there is a reindexing over it.

$$\begin{array}{ccc} s^*X & \dashrightarrow & X \\ {\scriptstyle f'}\,\vdots & & \downarrow{\scriptstyle f} \\ t^*Y & \dashrightarrow & Y \end{array} \qquad\qquad \begin{array}{ccc} A' & \xrightarrow{s} & A \\ {\scriptstyle df'=k}\downarrow & & \downarrow{\scriptstyle df} \\ B' & \xrightarrow{t} & B \end{array}$$

The vertical dotted arrow is uniquely determined.

The double category of an operad

The reindexing is compatible with composition

Reindexing squares can be composed vertically
(as well as horizontally).

The double category of an operad

The reindexing is compatible with composition

Reindexing squares can be composed vertically
(as well as horizontally).

The double category $\mathbb{D}_{\mathcal{O}}$

- The horizontal part of $\mathbb{D}_{\mathcal{O}}$ is (the domain of) the discrete family fibration on $\mathcal{O}_0$.

The double category of an operad

The reindexing is compatible with composition

Reindexing squares can be composed vertically
(as well as horizontally).

The double category $\mathbb{D}_{\mathcal{O}}$

- The horizontal part of $\mathbb{D}_{\mathcal{O}}$ is (the domain of) the discrete family fibration on $\mathcal{O}_0$.
- Vertical arrows (proarrows) are the maps of $\mathcal{D}_{\mathcal{O}}$, that is families of arrows in $\mathcal{O}$.

The double category of an operad

The reindexing is compatible with composition

Reindexing squares can be composed vertically
(as well as horizontally).

The double category $\mathbb{D}_{\mathcal{O}}$

- The horizontal part of $\mathbb{D}_{\mathcal{O}}$ is (the domain of) the discrete family fibration on $\mathcal{O}_0$.
- Vertical arrows (proarrows) are the maps of $\mathcal{D}_{\mathcal{O}}$, that is families of arrows in $\mathcal{O}$.
- Cells are the reindexing of families of arrows.

Operads as discrete fibrations

The existence and uniqueness of reindexing
of objects along mappings and of proarrows along pullbacks
can be expressed by saying that the indexing double functor
$d : \mathbb{D}_{\mathcal{O}} \to \mathbb{P}\mathrm{b}\,(\mathrm{Set}_{\mathrm{f}})$ is a discrete double fibration.

discrete double fibration (Lambert, 2021)

That is, both the components
$d_0 : \mathbb{D}_0 \to \mathsf{Set}_f$ and $d_1 : \mathbb{D}_1 \to \mathrm{Pb}(\mathsf{Set}_f)$
are discrete fibrations.

Operads as discrete fibrations

Lastly, $d : \mathbb{D} \to \mathbb{P}\mathrm{b}\,(\mathrm{Set_f})$
should satisfy the glueing conditions:

Glueing condition for objects

If X and Y are objects in $\mathbb{D}$ over A and B respectively,
there is a unique object Z over a sum $C = A + B$ in Set_f
which restricts to X and Y along injections.

Glueing condition for maps

If f and g are maps over s and t respectively,
there is a unique map h over a sum $r = s + t$ in Set_f^2
which restricts to f and g along injections
(which are pullbacks in Set_f).

Operads as discrete fibrations

Objects are families of objects...

The glueing condition for objects
assures that the horizontal part d^h of $d : \mathbb{D} \to \mathbb{P}\mathrm{b}\,(\mathrm{Set}_\mathrm{f})$
is indeed the family fibration on $\mathcal{O}_0$
(where $\mathcal{O}_0$ is the fiber over a terminal set).

...and maps are families of arrows

The glueing condition for maps
assures that a proarrow in $\mathbb{D}$ (that is, an object in $\mathbb{D}_1$)
is indeed a family of "single arrows", that is of proarrows
with the codomain indexed by a terminal set.

Main definition

We so arrive to our two lines definition of operad:

Non-skeletal notion of operad

An operad is a double discrete fibration $d : \mathbb{D} \to \mathbb{P}\mathrm{b}\,(\mathrm{Set_f})$ satisfying the glueing conditions.

Main definition

We so arrive to our two lines definition of operad:

Non-skeletal notion of operad

An operad is a double discrete fibration $d : \mathbb{D} \to \mathbb{P}\mathrm{b}\,(\mathrm{Set}_{\mathrm{f}})$ satisfying the glueing conditions.

Note that $\mathbb{D}$ is a strict double category,
and that $d : \mathbb{D} \to \mathbb{P}\mathrm{b}\,(\mathrm{Set}_{\mathrm{f}})$ is a strict double functor.

The category of operads

This notion of non-skeletal operad
is essentially equivalent to the classical one.

Morphisms $\mathcal{O} \to \mathcal{O}'$ of non-skeletal operads
are double functors $\mathbb{D}_{\mathcal{O}} \to \mathbb{D}_{\mathcal{O}'}$ over $\mathbb{P}\mathrm{b}\,(\mathsf{Set}_{\mathrm{f}})$.
The category of non-skeletal operads is equivalent
to the category of classical operads.

Operads as discrete fibrations (advantages)

Compatibility of permutation actions with composition

Figure from Leinster's book.

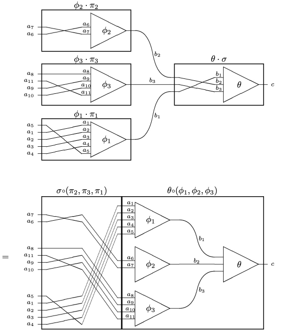


Figure 2-H: Symmetric multicategory axiom

$$\begin{aligned} & (\theta \cdot \sigma)\circ(\phi_{\sigma(1)} \cdot \pi_{\sigma(1)}, \ldots, \phi_{\sigma(n)} \cdot \pi_{\sigma(n)}) \\ = \; & (\theta\circ(\phi_1, \ldots, \phi_n)) \cdot (\sigma\circ(\pi_{\sigma(1)}, \ldots, \pi_{\sigma(n)})) \end{aligned}$$

Operads as discrete fibrations (advantages)

Confronting two ways of expressing compatibility

In our context, compatibility is given by vertical composition of cells.

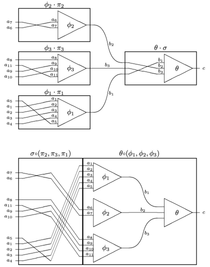
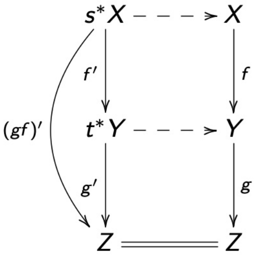


Figure 2-H: Symmetric multicategory axiom



Operads as discrete fibrations (advantages)

Confronting two ways of expressing compatibility

In our context, compatibility is given by
vertical composition of cells.

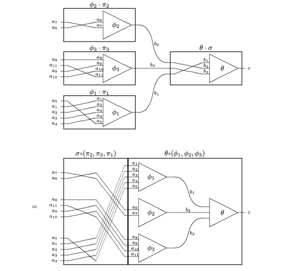


Figure 2-H: Symmetric multicategory axiom

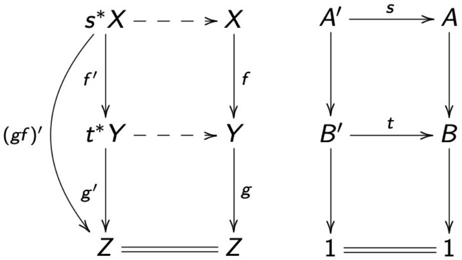


Table of Contents

Operads as double functors

Double Grothendieck correspondence (Lambert 2021, Paré 2011)

Double discrete fibrations $d : \mathbb{D} \to \mathbb{A}$ correspond to
lax functors $F : \mathbb{A}^{\mathrm{op}} \to \mathbb{S}\mathrm{et}$
to the (non-strict) double category of mappings and spans.

Operads as double functors

Double Grothendieck correspondence (Lambert 2021, Paré 2011)

Double discrete fibrations $d : \mathbb{D} \to \mathbb{A}$ correspond to lax functors $F : \mathbb{A}^{\mathrm{op}} \to \mathbb{S}\mathrm{et}$ to the (non-strict) double category of mappings and spans.

Universal property of the monoid construction (Cruttwell & Shulman 2010)

Since the monoid construction on $\mathbb{S}\mathrm{pan}$ gives $\mathbb{C}\mathrm{at}$, the double category of functors and profunctors, lax functors $F : \mathbb{A}^{\mathrm{op}} \to \mathbb{S}\mathrm{et}$ correspond to normal lax functors $F' : \mathbb{A}^{\mathrm{op}} \to \mathbb{C}\mathrm{at}$.

Operads as double functors

Thus, given a non-skeletal operad

$$d_{\mathcal{O}} : \mathbb{D} \to \mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}}$$

there are corresponding lax functors

$$F_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{S}\mathrm{et}$$

$$F'_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{C}\mathrm{at}$$

Operads as double functors

Furthermore it is easy to see that the glueing condition on $d_{\mathcal{O}}$ corresponds to the fact that $F_{\mathcal{O}}$ and $F'_{\mathcal{O}}$ preserve products.

Products in $(\mathbb{P}\mathrm{b}\,\mathrm{Set}_{\mathrm{f}})^{\mathrm{op}}$ are sums in $\mathbb{P}\mathrm{b}\,\mathrm{Set}_{\mathrm{f}}$,
that is pair of commuting squares whose horizontal sides are sums in Set_f (since Set_f is extensive).

$$\begin{array}{ccccc} A_1 & \xrightarrow{i} & A_1 + A_2 & \xleftarrow{j} & A_2 \\ {\scriptstyle s}\downarrow & & \downarrow{\scriptstyle s+t} & & \downarrow{\scriptstyle t} \\ B_1 & \xrightarrow[i]{} & B_1 + B_2 & \xleftarrow[j]{} & B_2 \end{array}$$

What is an operad?

Summarizing

A (non-skeletal) operad $\mathcal{O}$
can be defined in three equivalent ways:

1. A double discrete fibration with glueing
   $d_{\mathcal{O}} : \mathbb{D} \to \mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}}$.
2. A product-preserving lax functor
   $F_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{S}\mathrm{et}$.
3. A product-preserving normal lax functor
   $F'_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{C}\mathrm{at}$.

What is an operad?

Summarizing

A (non-skeletal) operad $\mathcal{O}$
can be defined in three equivalent ways:

1. A double discrete fibration with glueing
   $d_{\mathcal{O}} : \mathbb{D} \to \mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}}$.
2. A product-preserving lax functor
   $F_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{S}\mathrm{et}$.
3. A product-preserving normal lax functor
   $F'_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{C}\mathrm{at}$.

Each definition gives a different point of view
best suited to treat some aspects of operads.

Operads as double functors (explicitly)

The functor $F_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{S}\mathrm{et}$
takes a set $A \in \mathsf{Set}_f$ to the set $\mathcal{O}_0^A$,
and a mapping $t : A \to B$ to the span
whose vertex is formed by all families of arrows over t
and whose legs are given by domain and codomain.

The functor $F'_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{C}\mathrm{at}$
takes a set $A \in \mathsf{Set}_f$ to the category $\mathcal{O}_1^A$,
(where $\mathcal{O}_1$ is the category of unary arrows in $\mathcal{O}$)
and a mapping $t : A \to B$ to the profunctor $\overline{t}$ such that
$\overline{t}(X, Y)$ is formed by all families of arrows $f : X \to Y$ over t.

Table of Contents

Special operads

“horizontal triviality”

The horizontal part $d^h_{\mathcal{O}} : \mathbb{D}_0 \to \mathrm{Set}_f$ of an operad is forced to be the discrete family fibration on the set $\mathcal{O}_0$ (by the glueing or product preserving conditions).

“vertical peculiarity”

Thus, the character of $\mathcal{O}$ is in a sense determined by the vertical part $d^v_{\mathcal{O}} : \mathcal{D} \to \mathrm{Set}_f$.

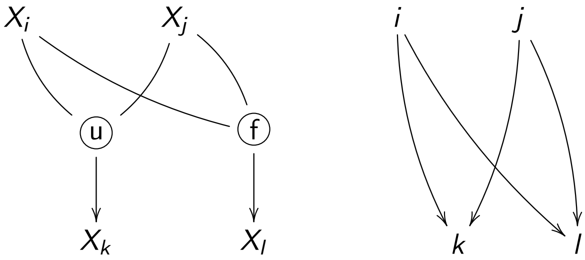
Symmetric monoidal categories

The vertical part $d^v_{\mathcal{O}} : \mathcal{D} \to \mathsf{Set}_f$ is an opfibration
if and only if $\mathcal{O}$ has tensor products.
That is, it is a symmetric monoidal category
in its universal form
(the representable multicategories of Hermida and Leinster).

Symmetric monoidal categories

Universal arrows

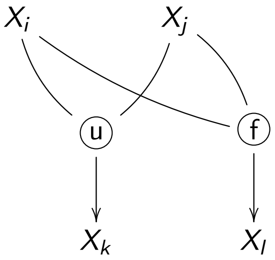
The opcartesian arrows for $d^{v}_{\mathcal{O}}$ are
the universal arrows defining tensor products.

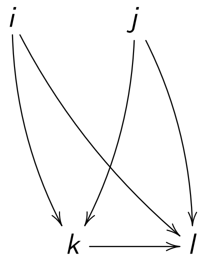


Symmetric monoidal categories

Universal arrows

The opcartesian arrows for $d^v_{\mathcal{O}}$ are
the universal arrows defining tensor products.

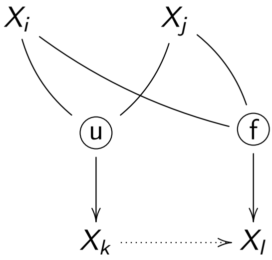


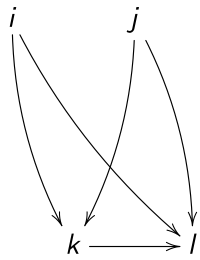


Symmetric monoidal categories

Universal arrows

The opcartesian arrows for $d^{v}_{\mathcal{O}}$ are
the universal arrows defining tensor products.

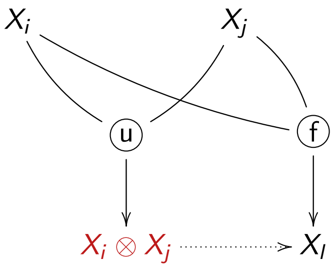


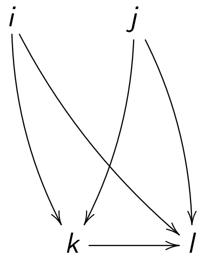


Symmetric monoidal categories

Universal arrows

The opcartesian arrows for $d^{\vee}_{\mathcal{O}}$ are
the universal arrows defining tensor products.



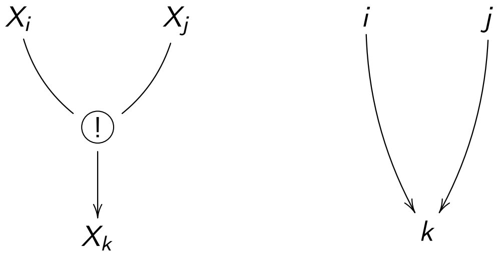


Commutative monoids

The vertical part $d^v_{\mathcal{O}} : \mathcal{D} \to \mathsf{Set}_f$ is a discrete opfibration
if and only if $\mathcal{O}$ is a commutative monoid.
That is, it is a discrete symmetric monoidal category.

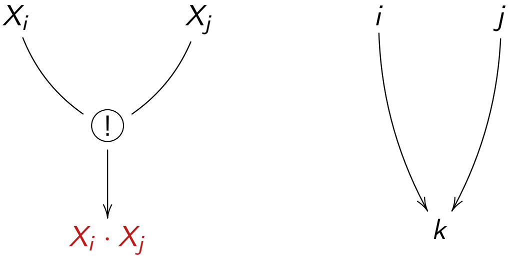
Commutative monoids

There is exactly one arrow out of any family of objects
(over a given mapping in Set_f)
whose codomain is the product of the family.



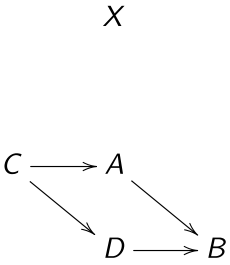
Commutative monoids

There is exactly one arrow out of any family of objects
(over a given mapping in Set_f)
whose codomain is the product of the family.



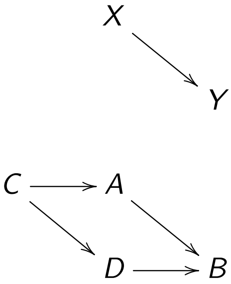
Commutative monoids

In elementary terms, a commutative monoid consists of
a discrete family fibration and a discrete opfibration
over finite sets, with the same objects which are compatible:



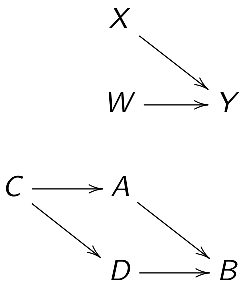
Commutative monoids

In elementary terms, a commutative monoid consists of
a discrete family fibration and a discrete opfibration
over finite sets, with the same objects which are compatible:



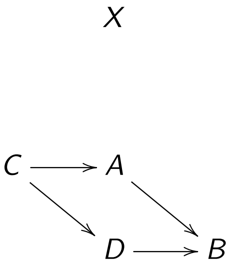
Commutative monoids

In elementary terms, a commutative monoid consists of
a discrete family fibration and a discrete opfibration
over finite sets, with the same objects which are compatible:



Commutative monoids

In elementary terms, a commutative monoid consists of
a discrete family fibration and a discrete opfibration
over finite sets, with the same objects which are compatible:



Commutative monoids

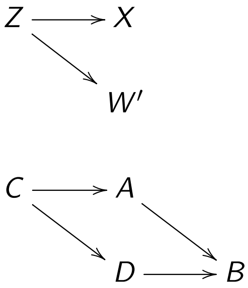
In elementary terms, a commutative monoid consists of
a discrete family fibration and a discrete opfibration
over finite sets, with the same objects which are compatible:

$$Z \longrightarrow X$$

$$\begin{array}{ccccc} C & \longrightarrow & A & & \\ & \searrow & & \searrow & \\ & & D & \longrightarrow & B \end{array}$$

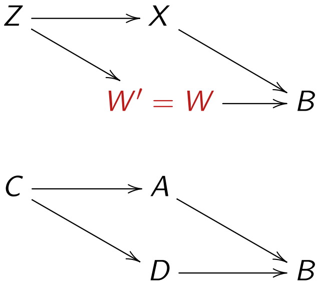
Commutative monoids

In elementary terms, a commutative monoid consists of
a discrete family fibration and a discrete opfibration
over finite sets, with the same objects which are compatible:



Commutative monoids

In elementary terms, a commutative monoid consists of
a discrete family fibration and a discrete opfibration
over finite sets, with the same objects which are compatible:



Commutative monoids as double functors

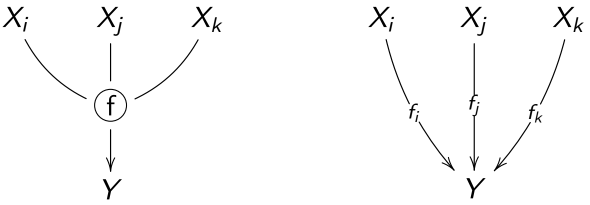
Corollary

A commutative monoid consists of a product-preserving strict double functor

$$(\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{S}\mathrm{q}\,\mathsf{Set}$$

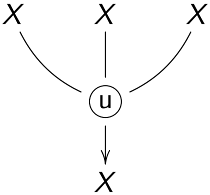
Sequential operads

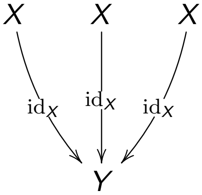
The vertical part $d_{\mathcal{O}}^{v} : \mathcal{D} \to \mathsf{Set}_f$ is a fibration
if and only if $\mathcal{O}$ is a sequential operad.



Sequential operads

The vertical part $d^v_{\mathcal{O}} : \mathcal{D} \to \mathsf{Set}_f$ is a fibration
if and only if $\mathcal{O}$ is a sequential operad.





The cartesian arrows in $\mathcal{D}$ are those made up
of identities (or isomorphims) in $\mathcal{C}$.

They form a "central monoid" in the operad,
which is in fact a way to characterize sequential operads (P. 2014).

Cocartesian monoidal categories

Corollary

The vertical part $d^v_{\mathcal{O}} : \mathcal{D} \to \mathsf{Set}_f$ is a bifibration
if and only if $\mathcal{O}$ is both monoidal and sequential.
That is, $\mathcal{O}$ is a cocartesian monoidal category.

Cocartesian monoidal categories

Corollary

The vertical part $d^v_{\mathcal{O}} : \mathcal{D} \to \mathsf{Set}_f$ is a bifibration
if and only if $\mathcal{O}$ is both monoidal and sequential.
That is, $\mathcal{O}$ is a cocartesian monoidal category.

Copying and deleting

The well-known characterization of cartesian monoidal categories
is a manifestation of (the dual of) the above fact:
the "copying-deleting" arrows are the cartesian maps of $d^v_{\mathcal{O}}$.

Cocartesian monoidal categories

Corollary

The vertical part $d^v_{\mathcal{O}} : \mathcal{D} \to \mathsf{Set}_f$ is a bifibration
if and only if $\mathcal{O}$ is both monoidal and sequential.
That is, $\mathcal{O}$ is a cocartesian monoidal category.

Copying and deleting

The well-known characterization of cartesian monoidal categories
is a manifestation of (the dual of) the above fact:
the "copying-deleting" arrows are the cartesian maps of $d^v_{\mathcal{O}}$.

Caution

The term "cartesian" is overworked:
cartesian arrow (of a fibration), cartesian monoidal category,
cartesian operad (to be considered later on)...

Exponentiable operads

An operad $d_{\mathcal{O}} : \mathbb{D} \to \mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}}$, is exponentiable
if and only if its vertical part $d^v : \mathcal{D} \to \mathsf{Set}_f$
is itself exponentiable in $\mathsf{Cat}/\mathsf{Set}_f$.

Exponentiable operads

An operad $d_{\mathcal{O}} : \mathbb{D} \to \mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}}$, is exponentiable
if and only if its vertical part $d^v : \mathcal{D} \to \mathsf{Set}_f$
is itself exponentiable in $\mathsf{Cat}/\mathsf{Set}_f$.

That is, d^v is a Conduché fibration.
These include fibrations and opfibrations, so that
symmetric monoidal categories and sequential operad
are both exponentiable.

Exponentiable operads

An operad $d_{\mathcal{O}} : \mathbb{D} \to \mathbb{P}\mathrm{b}\,\mathrm{Set}_{\mathrm{f}}$, is exponentiable
if and only if its vertical part $d^v : \mathcal{D} \to \mathrm{Set}_f$
is itself exponentiable in $\mathrm{Cat}/\mathrm{Set}_f$.

That is, d^v is a Conduché fibration.
These include fibrations and opfibrations, so that
symmetric monoidal categories and sequential operad
are both exponentiable.

Exponentiable operads coincide with
promonoidal symmetric multicategories.

Monoidal and exponentiable operads as double functors

An operad $F'_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set}_{\mathrm{f}})^{\mathrm{op}} \to \mathbb{C}\mathrm{at}$

is a monoidal
respectively,
is exponentiable

if and only if its vertical part $F^{v}_{\mathcal{O}} : \mathsf{Set}_f \to \mathsf{Prof}$,
(in general, a lax functor of bicategories)

lands in representable profunctors
respectively,
is a pseudofunctor.

Table of Contents

Changing the base

A natural generalization of this notion of operad is to replace Set_f with another base category $\mathcal{S}$.

Changing the base

A natural generalization of this notion of operad is to replace Set_f with another base category $\mathcal{S}$.

The first obvious idea is to use Set in place Set_f.

Infintary operads

1. A double discrete fibration with glueing $d_{\mathcal{O}} : \mathbb{D} \to \mathbb{P}\mathrm{b}\,\mathsf{Set}$.
2. A product-preserving lax functor $F_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set})^{\mathrm{op}} \to \mathbb{S}\mathrm{et}$.
3. A product-preserving normal lax functor $F'_{\mathcal{O}} : (\mathbb{P}\mathrm{b}\,\mathsf{Set})^{\mathrm{op}} \to \mathbb{C}\mathrm{at}$.

Changing the base: example

The infinitary sequential operad on $\mathcal{C}$

The usual family fibration $d : \mathrm{Fam}\,\mathcal{C} \to \mathsf{Set}$
is the vertical part of an infinitary operad $\mathcal{C}^{\infty}_{\blacktriangleright}$.

Changing the base: example

The infinitary sequential operad on $\mathcal{C}$

The usual family fibration $d : \mathrm{Fam}\,\mathcal{C} \to \mathsf{Set}$
is the vertical part of an infinitary operad $\mathcal{C}^{\infty}_{\blacktriangleright}$.

If $\mathcal{C}$ has small sums the family fibration is a bifibration,
giving an instance of infinitary monoidal category.

Changing the base: example

The infinitary sequential operad on $\mathcal{C}$

The usual family fibration $d : \mathrm{Fam}\,\mathcal{C} \to \mathsf{Set}$
is the vertical part of an infinitary operad $\mathcal{C}^{\infty}_{\blacktriangleright}$.

If $\mathcal{C}$ has small sums the family fibration is a bifibration,
giving an instance of infinitary monoidal category.

Isomorphism classes of an infinitary monoidal category
give an infinitary commutative monoid.

Changing the base: example

The infinitary sequential operad on $\mathcal{C}$

The usual family fibration $d : \mathrm{Fam}\,\mathcal{C} \to \mathsf{Set}$
is the vertical part of an infinitary operad $\mathcal{C}^{\infty}_{\blacktriangleright}$.

If $\mathcal{C}$ has small sums the family fibration is a bifibration,
giving an instance of infinitary monoidal category.

Isomorphism classes of an infinitary monoidal category
give an infinitary commutative monoid.

This is a way to make it precise the idea that
universal sums or products can be “decategorified”
to give a monoid-like structure, not only in the finite case.

Cartesian operads

The notion of cartesian operad can be developed on any category $\mathcal{S}$ is with pullbacks.

Cartesian operads

The notion of cartesian operad can be developed
on any category $\mathcal{S}$ is with pullbacks.

Idea

operads : symmetric monoidal $=$?? : cartesian monoidal

minimum requirement

monoidal cartesian operads $=$ cartesian monoidal categories.

a notion of algebraic theory

alternative to (and more flexible than) Lawvere theories.

Cartesian operads

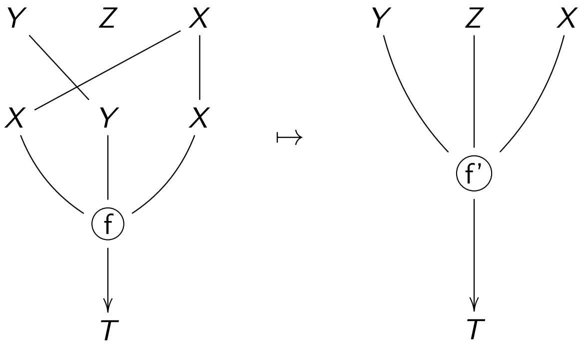
weakening and contraction in cartesian monoidal categories

$$f : X, Y, X \to T \quad \mapsto \quad f' : Y, Z, X \to T$$

$$f'(y, z, x) = f(x, y, x)$$

Cartesian operads : "contraction" and "weakening"

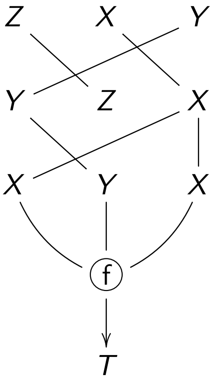
The reindexing of the domain acts covariantly on f to give f'.



Cartesian operads

Reindexing arrows act on maps.

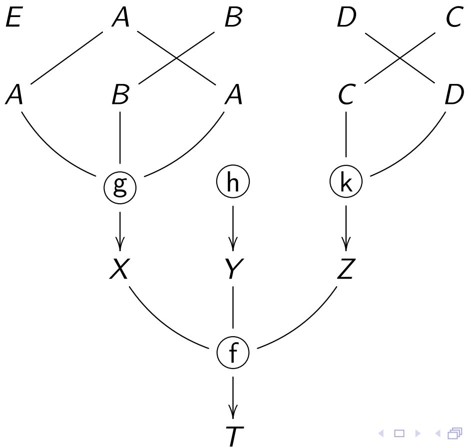
This is unambiguous:



Cartesian operads

The action is compatible with composition from below.

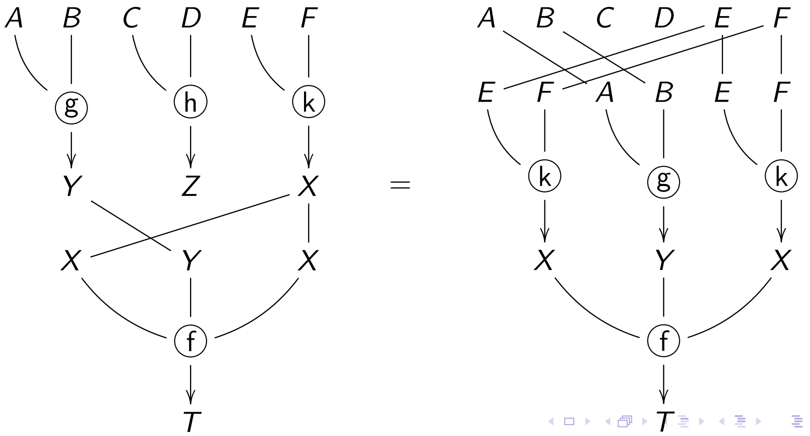
This is unambiguous:



Cartesian operads

Combing

The action is compatible with composition from above.

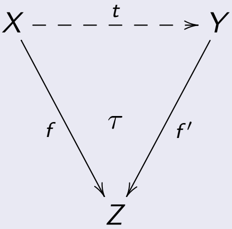


Cartesian operads

Cartesian operads on $\mathcal{S}$

is an operad $d_{\mathcal{O}} : \mathbb{D} \to \mathbb{P}\mathrm{b}\,\mathcal{S}$,
such that $\mathbb{D}$ has also triangular cells.

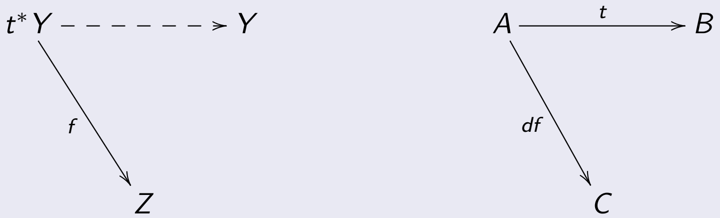
Triangular cells give the covariant reindexing



Cartesian operads: axioms

Covariant reindexing of maps

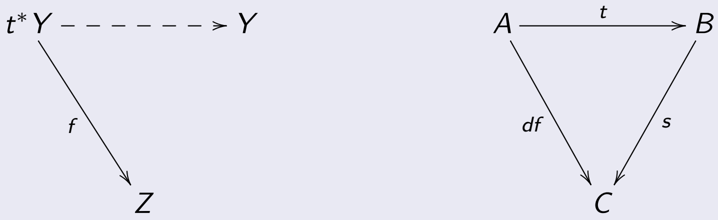
Given a proarrow $f : t^*Y \to Z$ in $\mathcal{D}$,
and a commutative triangle in $\mathcal{S}$ completing df and t,
there is a unique extension to a triangular cell over it:



Cartesian operads: axioms

Covariant reindexing of maps

Given a proarrow $f : t^*Y \to Z$ in $\mathcal{D}$,
and a commutative triangle in $\mathcal{S}$ completing df and t,
there is a unique extension to a triangular cell over it:



Cartesian operads: axioms

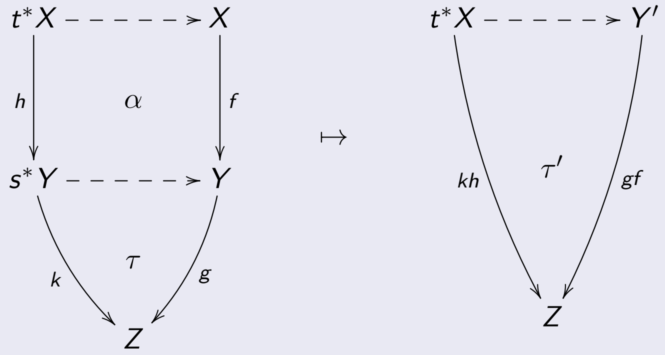
Covariant reindexing of maps

Given a proarrow $f : t^*Y \to Z$ in $\mathcal{D}$,
and a commutative triangle in $\mathcal{S}$ completing df and t,
there is a unique extension to a triangular cell over it:

Cartesian operads: axioms

A triangular cell can be pasted with a square cell,
giving a triangular cell.

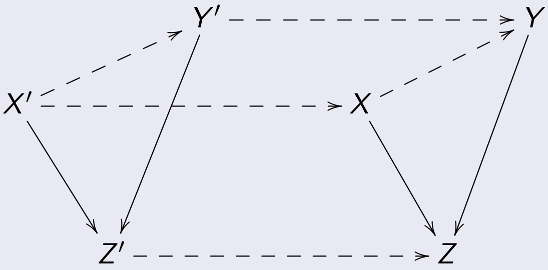
Pasting = Combing



Cartesian operads: axioms

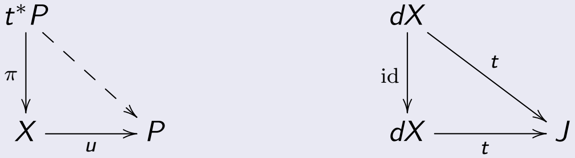
Triangular cells are stable with respect to reindexing.

Stability



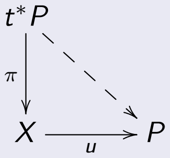
Algebraic products

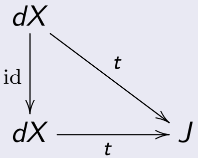
An algebraic product of X along t is a P with maps π and u...



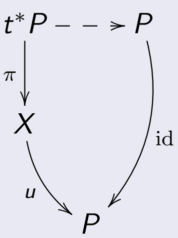
Algebraic products

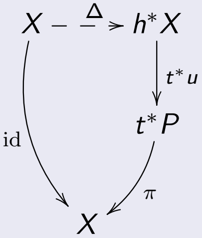
An algebraic product of X along t is a P with maps π and u...





...such that the following are both triangular cells:





Main result

Main result for cartesian operads

For a cartesian operad $\mathcal{O}$ on $\mathcal{S}$, the following are equivalent:

1. $\mathcal{O}$ has algebraic products.
2. $\mathcal{O}$ has universal products.
3. $\mathcal{O}$ is monoidal (representable).

This result indicates that we have indeed captured a proper notion of cartesian operad.

Cartesian + Sequential = Semiadditive

Further evidence

One can also generalize results such as the following:

Cartesian + Sequential = Semiadditive (P. 2014)

Cartesian structures on sequential operads
correspond to enrichments of the underlying category
in the category of commutative monoids.

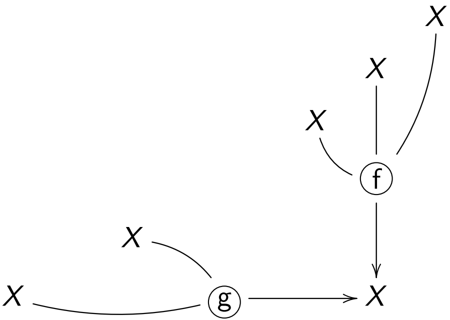
In the present context, objects are to be intended as sections $x : \mathcal{C} \to \mathbb{D}^h$ of $d^h_{\mathcal{O}}$, and the commutative monoid $\mathcal{O}(x; y)$ is a commutative monoid on $\mathcal{S}$ in the generalized sense.

Commuting internal operations

One important notion that can be considered in operads
and that can be expressed most naturally
in the present context of operads on $\mathcal{S}$
is that of commuting internal operations
(that is, arrows involving just one object).

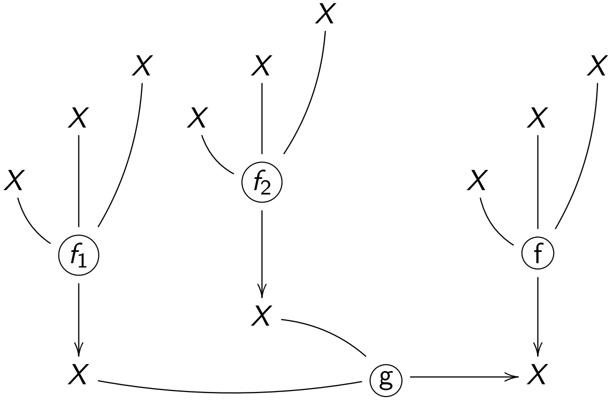
Commuting internal operations

Two internal operations with the same codomain.



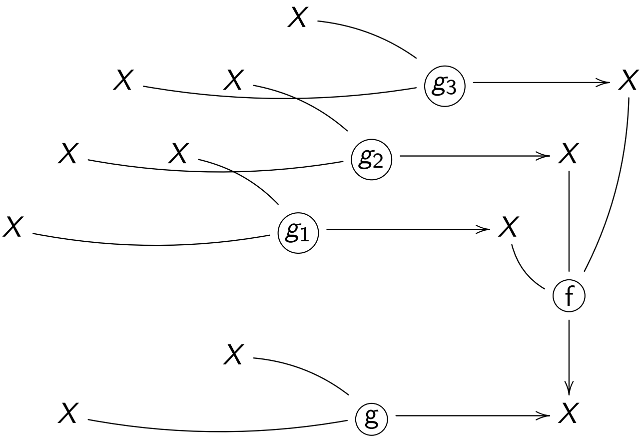
Commuting internal operations

Reindexing f along dg.



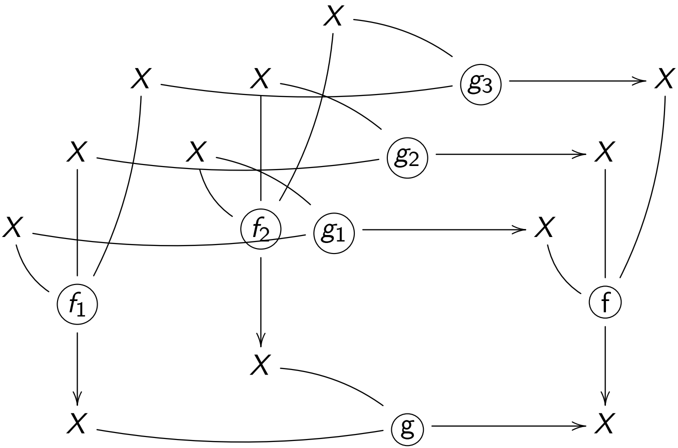
Commuting internal operations

Reindexing g along df.



Commuting internal operations

The two reindexing can be composed and may give the same result.



Commuting internal operations

Commuting internal operations

Two internal operations f and g, with the same codomain, commute if the square below commutes in $\mathcal{D}$: $fg' = gf'$ where f' and g' are the reindexing f and g.

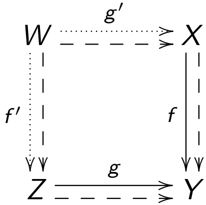


Table of Contents

Fibrations as discrete double fibrations

Decoupled fibrations

The present approach to operads also suggests
the more general notion of a decoupled fibration,
that is a discrete double fibration $d : \mathbb{D} \to \mathbb{S}\mathrm{q}\,\mathcal{S}$.

The idea is that such a d can be seen as a fibration
where the cartesian arrows become the horizontal arrows
and are separated from the other arrows (the vertical ones).

Indeed, we have the following result:

Split fibrations $d : \mathcal{D} \to \mathcal{S}$ coincide with
discrete double fibrations $d : \mathbb{D} \to \mathbb{S}\mathrm{q}\,\mathcal{S}$
such that $\mathbb{D}$ has companions preserved by d.

To explore

Another promising development is to consider
operads on double categories
which are more "relations-like",
for instance cospans in Set_f.

In this case, it seems appropriate to consider,
as cells in the base double category,
summand squares rather than pullback squares.
(There is no difference if $\mathcal{S}$ is extensive.)

Conclusions

We have presented a non-skeletal approach to operads.
The main advantages are:

- It avoids the introduction of spurious orders, rendering neater the notion.
- We can exploit the language of double categories, to capture in a smooth way various classes of operads and to highlight their connections.
- We can usefully generalize the base category.

Conclusions

We have presented a non-skeletal approach to operads.
The main advantages are:

- It avoids the introduction of spurious orders, rendering neater the notion.
- We can exploit the language of double categories, to capture in a smooth way various classes of operads and to highlight their connections.
- We can usefully generalize the base category.

THANKS!