Computads for Weak ω-Categories as an Inductive Type

Christopher J. Dean Eric Finster Ioannis Markakis [1]
David Reutter Jamie Vicary

TACL 2022

UNIVERSITY OF CAMBRIDGE

[1] The speaker is being partially supported by the Onassis Foundation - Scholarship ID: F ZQ 039-1/2020-2021

ONASSIS FOUNDATION

Overview

1. Globular Sets and Trees
2. Computads and weak ω-categories
3. Consequences of our Definition

Globular Sets

Definition

The category $\mathbb{G}$ of globes has objects natural numbers and morphisms generated by cosource and cotarget inclusions $s, t : [n] \to [n+1]$ under the globularity relations

$$s \circ s = t \circ s \qquad\qquad s \circ t = t \circ t$$

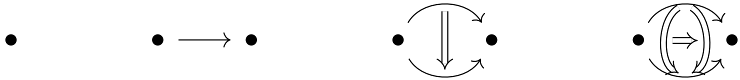
Globular Sets

Definition

The category $\mathbb{G}$ of globes has objects natural numbers and morphisms generated by cosource and cotarget inclusions $s, t : [n] \to [n+1]$ under the globularity relations

$$s \circ s = t \circ s \qquad s \circ t = t \circ t$$

- The category of *globular sets* is the presheaf category $[\mathbb{G}^{\mathrm{op}}, \mathrm{Set}]$
- For a globular set X, we call elements of X_n the n-cells of X and visualise them (with their sources and targets) as follows:



Strict ω-categories

Definition

A *strict ω-category* is a globular set X equipped with composition and identity operations

$$\circ_k : X_n \times_{X_k} X_n \to X_n \qquad (0 \leq k < n)$$
$$\mathrm{id} : X_n \to X_{n+1} \qquad (n \in \mathbb{N})$$

satisfying certain associativity, unitality and interchange laws.

Strict ω-categories

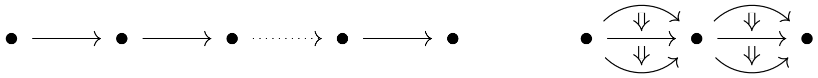
Definition

A *strict ω-category* is a globular set X equipped with composition and identity operations

$$\circ_k : X_n \times_{X_k} X_n \to X_n \qquad (0 \leq k < n)$$
$$\mathrm{id} : X_n \to X_{n+1} \qquad (n \in \mathbb{N})$$

satisfying certain associativity, unitality and interchange laws.

It follows by the definition of strict ω-categories that certain diagrams of cells, can be composed in a unique way.



Batanin Trees

Definition

The set Tree of *Batanin trees* is generated inductively by one constructor

$$\mathrm{bt} : \mathrm{List}(\mathrm{Tree}) \to \mathrm{Tree}$$

- We also construct inductively for every Batanin tree B a finite globular sets $\mathrm{Pos}\, B$, whose cells we call positions.
- Those globular sets are also known as *globular sums*.
- They will be the arities of the operations of our weak ω-categories.

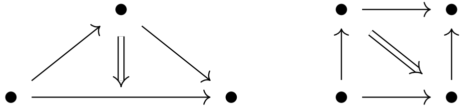
Example

The disks and composable 1-arrows may be defined inductively as follows:

$$D_0 = \mathrm{bt}[] \qquad D_{n+1} = \mathrm{bt}[D_n] \qquad B_{1,n} = \mathrm{bt}[D_0, D_0, \ldots, D_0]$$

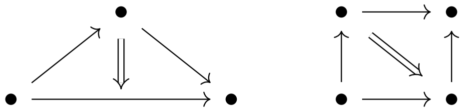
Computads for weak ω-categories

Every globular set freely generates a (strict / weak) ω-category, but we can also generate ω-categories by more general shapes



Computads for weak ω-categories

Every globular set freely generates a (strict / weak) ω-category, but we can also generate ω-categories by more general shapes

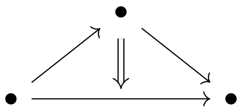


We will define for every $n \in \mathbb{N}$ a category Comp_n of n-computads together with

- a truncation functor $u_n : \mathrm{Comp}_n \to \mathrm{Comp}_{n-1}$ (for $n > 0$),

Computads for weak ω-categories

Every globular set freely generates a (strict / weak) ω-category, but we can also generate ω-categories by more general shapes

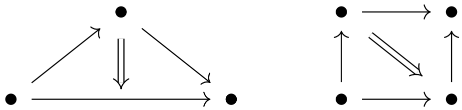


We will define for every $n \in \mathbb{N}$ a category Comp_n of n-computads together with

- a truncation functor $u_n : \mathrm{Comp}_n \to \mathrm{Comp}_{n-1}$ (for $n > 0$),
- a functor $\mathrm{Cell}_n : \mathrm{Comp}_n \to \mathrm{Set}$,

Computads for weak ω-categories

Every globular set freely generates a (strict / weak) ω-category, but we can also generate ω-categories by more general shapes

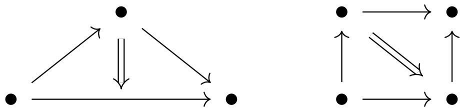


We will define for every $n \in \mathbb{N}$ a category Comp_n of n-computads together with

- a truncation functor $u_n : \mathsf{Comp}_n \to \mathsf{Comp}_{n-1}$ (for $n > 0$),
- a functor $\mathsf{Cell}_n : \mathsf{Comp}_n \to \mathsf{Set}$,
- natural transformations $\mathsf{src}, \mathsf{tgt} : \mathsf{Cell}_n \Rightarrow \mathsf{Cell}_{n-1}\, u_{n-1}$ (for $n > 0$),

Computads for weak ω-categories

Every globular set freely generates a (strict / weak) ω-category, but we can also generate ω-categories by more general shapes

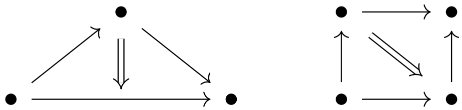


We will define for every $n \in \mathbb{N}$ a category Comp_n of n-computads together with

- a truncation functor $u_n : \mathrm{Comp}_n \to \mathrm{Comp}_{n-1}$ (for $n > 0$),
- a functor $\mathrm{Cell}_n : \mathrm{Comp}_n \to \mathrm{Set}$,
- natural transformations $\mathrm{src}, \mathrm{tgt} : \mathrm{Cell}_n \Rightarrow \mathrm{Cell}_{n-1}\, u_{n-1}$ (for $n > 0$),
- a distinguished n-computad Pd_n^B for every Batanin tree B,

Computads for weak ω-categories

Every globular set freely generates a (strict / weak) ω-category, but we can also generate ω-categories by more general shapes



We will define for every $n \in \mathbb{N}$ a category Comp_n of n-computads together with

- a truncation functor $u_n : \mathrm{Comp}_n \to \mathrm{Comp}_{n-1}$ (for $n > 0$),
- a functor $\mathrm{Cell}_n : \mathrm{Comp}_n \to \mathrm{Set}$,
- natural transformations $\mathrm{src}, \mathrm{tgt} : \mathrm{Cell}_n \Rightarrow \mathrm{Cell}_{n-1}\, u_{n-1}$ (for $n > 0$),
- a distinguished n-computad Pd_n^B for every Batanin tree B,
- a set $\mathrm{Full}_n(B)$ of pairs of parallel cells of Pd_n^B.

Computads and Homomorphisms

Definition

- A 0-computad is a set.
- An $(n+1)$-computad C consists of
  - an n-computad C_{n-1},
  - a set of generators V_{n+1}^{C},
  - a function ϕ_n^C assigning to each generator a pair of parallel cells of C_n.
- A computad $C = (C_n)_{n \in \mathbb{N}}$ consists of n-computad for every n such that $u_{n+1} C_{n+1} = C_n$.

Computads and Homomorphisms

Definition

- A 0-computad is a set.
- An $(n+1)$-computad C consists of
  - an n-computad C_{n-1},
  - a set of generators V^C_{n+1},
  - a function ϕ^C_n assigning to each generator a pair of parallel cells of C_n.
- A computad $C = (C_n)_{n \in \mathbb{N}}$ consists of n-computad for every n such that $u_{n+1} C_{n+1} = C_n$.

Definition

- A 0-homomorphism is a function.
- An $(n+1)$-homomorphism $C \to D$ consists
  - an n-homomorphism $C_n \to D_n$ and
  - a function $V^C_n \to \mathrm{Cell}_{n+1}\, D$ commuting with source and target.
- A homomorphism is a compatible sequence of n-homomorphisms.

The Cells of a Computad

Definition

- The set of cells of a 0-computad C is C.
- The set of cells of an $(n+1)$-computad C is generated inductively by
  - There exists a cell $(\mathrm{var}\, v)$ for every generator $v \in V_n^C$.
  - There exists a cell $\mathrm{coh}(B,(a,b),\tau)$ for every tree of dimension at most $n+1$, every full pair of parallel n-cells of it and every homomorphism $\tau : \mathrm{Pd}_{n+1}^B \to C$.

The source of positive dimensional cells are defined recursively by the following formulas, while the target is defined similarly

$$\mathrm{src}(\mathrm{var}\, v) = \mathrm{pr}_1\, \phi_n^C(v)$$
$$\mathrm{src}(\mathrm{coh}(B,(a,b),\tau)) = \mathrm{Cell}_{n-1}(u_n\tau)(a)$$

Remark

This construction defines a functor Cell : Comp → Glob.

The Pasting Diagrams

Definition

Let B a Batanin tree. The distinguished n-computad Pd^B_n is defined by

- $\mathsf{Pd}^B_0 = \mathsf{Pos}_0\, B$
- $\mathsf{Pd}^B_{n+1} = (\mathsf{Pd}^B_n, \mathsf{Pos}_{n+1}\, B, \phi^B_{n+1})$ where

$$\phi^B_{n+1}(p) = (\mathsf{var}\,\mathsf{src}\,p, \mathsf{var}\,\mathsf{tgt}\,p)$$

- $\mathsf{Pd}^B = (\mathsf{Pd}^B_n)_{n\in\mathbb{N}}$

We will say that a pair of parallel cells $(a, b) \in \mathsf{Cell}_n(B)$ is full when a and b cover the n-dimensional source and target of B respectively.

Remark

This construction generalises to an embedding $\mathsf{Cptd} : \mathsf{Glob} \to \mathsf{Comp}$.

Example: Low Dimensional Composition and Associators

- Consider the following two 1-computads

Example: Low Dimensional Composition and Associators

- Consider the following two 1-computads

Example: Low Dimensional Composition and Associators

- Consider the following two 1-computads

$$x \xrightarrow{f} y \xrightarrow{g} z$$

Figure: $\mathsf{Pd}^{B_{1,2}}$

Example: Low Dimensional Composition and Associators

- Consider the following two 1-computads

$$x \xrightarrow{f} y \xrightarrow{g} z$$

Figure: $\mathrm{Pd}^{B_{1,2}}$

$$x' \xrightarrow{f'} y' \xrightarrow{g'} z' \xrightarrow{h'} w'$$

Figure: $\mathrm{Pd}^{B_{1,3}}$

Example: Low Dimensional Composition and Associators

- Consider the following two 1-computads

$$x \xrightarrow{f} y \xrightarrow{g} z$$

Figure: $\mathsf{Pd}^{B_{1,2}}$

$$x' \xrightarrow{f'} y' \xrightarrow{g'} z' \xrightarrow{h'} w'$$

Figure: $\mathsf{Pd}^{B_{1,3}}$

- The pair (x, z) covers the 0-dimensional boundary of $B_{1,2}$, so we get a cell $g \circ f = \mathrm{coh}(B_{1,2}, (x, z), \mathrm{id})$.

Example: Low Dimensional Composition and Associators

- Consider the following two 1-computads

$$x \xrightarrow{f} y \xrightarrow{g} z$$

Figure: $\mathrm{Pd}^{B_{1,2}}$

$$x' \xrightarrow{f'} y' \xrightarrow{g'} z' \xrightarrow{h'} w'$$

Figure: $\mathrm{Pd}^{B_{1,3}}$

- The pair (x, z) covers the 0-dimensional boundary of $B_{1,2}$, so we get a cell $g \circ f = \mathrm{coh}(B_{1,2}, (x, z), \mathrm{id})$.
- By substitution, we may define cells $h' \circ (g' \circ f')$ and $(h' \circ g') \circ f'$ over $\mathrm{Pd}^{B_{1,3}}$.

Example: Low Dimensional Composition and Associators

- Consider the following two 1-computads

$$x \xrightarrow{f} y \xrightarrow{g} z$$

Figure: $\mathrm{Pd}^{B_{1,2}}$

$$x' \xrightarrow{f'} y' \xrightarrow{g'} z' \xrightarrow{h'} w'$$

Figure: $\mathrm{Pd}^{B_{1,3}}$

- The pair (x, z) covers the 0-dimensional boundary of $B_{1,2}$, so we get a cell $g \circ f = \mathrm{coh}(B_{1,2}, (x, z), \mathrm{id})$.
- By substitution, we may define cells $h' \circ (g' \circ f')$ and $(h' \circ g') \circ f'$ over $\mathrm{Pd}^{B_{1,3}}$.
- Both cells cover the 1-dimensional boundary of $B_{1,3}$, which is $B_{1,3}$ itself, so we have similarly a 2-cell

$$\mathrm{assoc}_{f',g',h'} : h' \circ (g' \circ f') \Rightarrow (h' \circ g') \circ f'$$

and one in the opposite direction.

ω-categories

Proposition

The functor $\mathsf{Cptd} : \mathsf{Glob} \to \mathsf{Comp}$ *is left adjoint to* Cell*.*

Definition

An ω-category is an algebra for $\mathsf{fc}^w = \mathsf{Cell}\,\mathsf{Cptd}$.

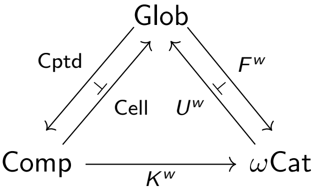
ω-categories

Proposition

The functor Cptd : Glob $\to$ Comp *is left adjoint to* Cell.

Definition

An ω-category is an algebra for $\mathrm{fc}^w = \mathrm{Cell}\,\mathrm{Cptd}$.



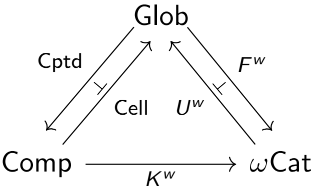
ω-categories

Proposition

The functor Cptd : Glob $\to$ Comp *is left adjoint to* Cell.

Definition

An ω-category is an algebra for $\mathrm{fc}^w = \mathrm{Cell}\,\mathrm{Cptd}$.



Proposition

Computads embed fully faithfully into ωCat.

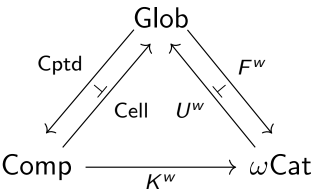
ω-categories

Proposition

The functor Cptd : Glob $\to$ Comp *is left adjoint to* Cell.

Definition

An ω-category is an algebra for $\mathrm{fc}^w = \mathrm{Cell}\,\mathrm{Cptd}$.



Proposition

Computads embed fully faithfully into ωCat.

Theorem

fc^w *is the initial contractible globular operad.*

The Variable-to-Variable Subcategory

- Morphisms of computads sending variables to variables form a subcategory $\mathsf{Comp}^{\mathsf{var}}$ of Comp.
- We inductively construct a family of computads $J(c)$ indexed by cells of the terminal computad such that

$$\mathsf{Cell}_n \cong \coprod_{c \in \mathsf{Cell}_n(1)} \mathsf{Comp}^{\mathsf{var}}(J(c), -)$$

- Let $\mathcal{V}$ the full subcategory of $\mathsf{Comp}^{\mathsf{var}}$ on the computads of the form $J(\mathsf{var}\ v)$.

Theorem

$$\mathsf{Comp}^{\mathsf{var}} \cong [\mathcal{V}^{\mathsf{op}}, \mathsf{Set}]$$

Computads are Cellular Objects

Theorem

The ω-category free on a computad $C = (C_n)$ is the colimit of the ones free on C_n, which fit in pushout squares of the form

$$\begin{array}{ccc} \coprod \mathbb{S}^{n-1} & \rightarrowtail & \coprod \mathbb{D}^n \\ \downarrow & & \downarrow \\ K^w C_{n-1} & \rightarrowtail & K^w C_n \end{array}$$

In particular, free ω-categories are cofibrant for $I = \{\mathbb{S}^{n-1} \hookrightarrow \mathbb{D}^n\}$.

- We can build a right adjoint $W : \omega\mathrm{Cat} \to \mathrm{Comp}^{\mathrm{var}}$ to the free computad functor, hence a comonad Q on ωCat.
- Using a recognition principle of Garner, we see that this is the universal cofibrant replacement comonad.

References

[1] M. A. Batanin. “Computads for finitary monads on globular sets”. In: *Higher category theory*. Vol. 230. Amer. Math. Soc., 1998.

[2] E. Finster and S. Mimram. “A type-theoretical definition of weak ω-categories”. In: *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. IEEE. 2017, pp. 1–12.

[3] R. Garner. “Homomorphisms of higher categories”. In: *Advances in Mathematics* 224.6 (Aug. 2010), pp. 2269–2311.

Thank you!