Electronic Notes in Theoretical Informatics And Computer Science

ENTICS
https://entics.episciences.org

Volume 4
Proceedings of MFPS 2024

CaTT Contexts are Finite Computads*

Thibaut Benjamin[a,1] Ioannis Markakis[a,2,4] Chiara Sarti[a,3]

[a] *Department of Computer and Technology*
University of Cambridge
Cambridge, United Kingdom

Abstract

Two novel descriptions of weak ω-categories have been recently proposed, using type-theoretic ideas. The first one is the dependent type theory `CaTT` whose models are ω-categories. The second is a recursive description of a category of computads together with an adjunction to globular sets, such that the algebras for the induced monad are again ω-categories. We compare the two descriptions by showing that there exits a fully faithful morphism of categories with families from the syntactic category of `CaTT` to the opposite of the category of computads, which gives an equivalence on the subcategory of finite computads. We derive a more direct connection between the category of models of `CaTT` and the category of algebras for the monad on globular sets, induced by the adjunction with computads.

Keywords: ω-categories, computads, dependent type theory, categories with families

1 Introduction

Higher categories have recently found applications in theoretical computer science and other fields. They describe the iterated structure of identity types in Martin-Löf type theory [1,23,30], a fact that has been instrumental in the emergence of homotopy type theory [29]. They are also crucial in higher-dimensional rewriting [26], homotopy theory [16,18] and homology theory [19]. There exists a wide range of flavours of higher categories, differentiated by the shapes of their higher morphisms and the ways that they are composed. For a review of the various definitions, we refer to the work of Leinster [21].

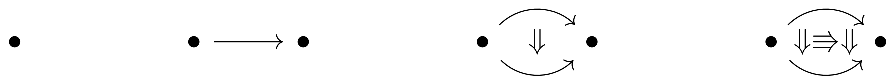
In this paper, we focus on globular weak ω-categories, a particular model of higher categories. Different definitions of weak ω-categories have been given by Batanin and Leinster [4,22], and by Grothendieck and Maltsiniotis [17,24]. More recently, two syntactic descriptions of weak ω-categories have been proposed. One takes the form of a dependent type theory called `CaTT`, introduced by Finster and Mimram [15]. The other one gives a description of the free ω-category monad using inductive types, which was introduced by Dean et al. [13] and is given via an adjunction to a category of *computads*. The equivalence between

* We would like to thank Jamie Vicary for his feedback.
[1] Email: tjb201@cam.ac.uk
[2] Email: ioannis.markakis@cl.cam.ac.uk
[3] Email: cs2197@cam.ac.uk
[4] The author would like to acknowledge funding by the Onassis Foundation - Scholarship ID: F ZQ 039-1/2020-2021

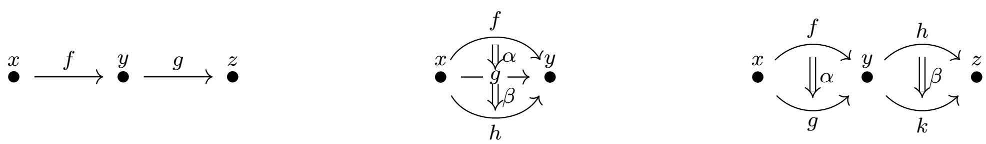
Published December 15, 2024
10.46298/entics.14675
Proceedings Available Online at https://doi.org/10.46298/entics.proceedings.mfps40

models of the type theory `CaTT` and algebras of the free ω-category monad of Dean et al. has been established by a chain of results. Benjamin, Finster and Mimram [7] show that models of `CaTT` are equivalent to Grothendieck-Maltsiniotis weak ω-categories. Dean et al. [13] show that their monad is the one of Leinster. Finally, the equivalence of the theories of Batanin-Leinster and Grothendieck-Maltsiniotis has been established by Ara [2] and Bourke [10]. In this paper, we establish a more direct comparison between the two syntactic descriptions by showing that the contexts of `CaTT` correspond to finite computads.

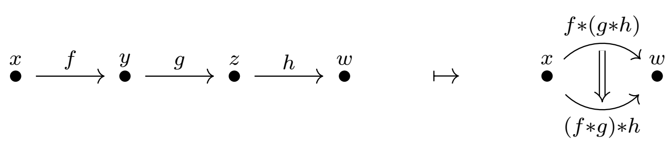
The notion of ω-categories is an extension of the notion of globular sets. The data of a globular sets is a set of cells of various dimensions, equipped with source and target operations. Those operations allows us to visualise cells as directed analogues of disks of various dimensions, as depicted below in low dimensions:



An ω-category is a globular set G equipped with ways to compose certain configurations of cells. Those operations generalise for instance the composition operations of ordinary categories and bicategories, in the sense that the following diagrams of cells admit a composite:



One may expect from ordinary category theory that those operations are strictly associative and unital. This leads to the notions of strict ω-categories. However, for an arbitrary ω-category, associativity and unitality only hold in a weaker sense. More precisely, there exist operations producing higher-dimensional *coherence* cells witnessing those axioms. For example, there is an operation, depicted below, producing a witness cell for the associativity of the composition of three 1-cells:



This leads to an infinite tower of operations, where higher witness cells relate composites of lower-dimensional witness cells. For example, there are cells relating the two sides of the pentagon and triangle equations of monoidal categories.

The dependent type theory `CaTT` gives a syntactic way to systematically track those operations. Types in `CaTT` combinatorially encode directed spheres, while terms encode directed disks. The terms are produced by a unique term constructor coh parametrised by a context Γ of a particular kind called a *pasting context* together with a type $\Gamma \vdash A$ satisfying some side conditions. Pasting contexts encode the arities of the composition and coherence operations of ω-categories, i.e. the configurations of cells that may be composed. When the side condition is satisfied, the term constructor $\mathrm{coh}_{\Gamma,A}$ produces a canonical inhabitant for the type A. This inhabitant can be seen as the composite cell of the variables of Γ, or a coherence cell relating the source and target of A, depending on the dimension of A. We note that this type theory has no equality between terms, due to the lack of equations between the operations of the ω-categories it models.

Contexts in the type theory `CaTT` are finite lists of generators for ω-categories. An alternative way to freely generate ω-categories, due to Street [27], Burroni [11] and Batanin [3], is that of *computads* or *polygraphs*. Computads are sets of generators stratified by dimension, together with attaching functions assigning to each cell a source and target, which are lower-dimensional cells of the free ω-category generated by the computad. More precisely, computads and the free ω-category on them are defined by mutual induction on their dimension. Inspired by the type theory `CaTT`, Dean et al. [13] recently gave a new presentation of the free ω-category on a computad using inductive types. In this presentation, the sets of cells of the free ω-category are inductively generated by a constructor coh with similar arguments as the term constructor of CaTT.

Our main contribution is showing that the syntactic category of CaTT is equivalent to the opposite of the category of finite computads. We do this by first equipping the opposite of the category of computads with a structure of category with families. We then establish a bijection between pasting contexts and *Batanin trees*, the arities of the constructor coh in the case of computads. We construct recursively a morphism of categories with families $R\colon \mathcal{S}_{\texttt{CaTT}} \to \mathrm{Comp}^{\mathrm{op}}$ from the syntactic category of CaTT to the opposite of the category of computads, comparing types A satisfying the side condition of the term constructor to the *full spheres* of the cell constructor. Finally, in our main Theorem 6.1, we shown that this morphism is fully faithful and essentially surjective on the subcategory of finite computads, establishing the claimed equivalence between contexts of CaTT and finite computads.

Overview of the paper.

In Section 2, we give an overview of categories with families and the dependent type theories GSeTT and CaTT. In Section 3, we present the definition of globular sets and computads, and the adjunction between them. Then in Section 4, we briefly describe the equivalence between the syntactic category of GSeTT and finite globular sets, established by Benjamin, Finster, Mimram [7]. In Section 5, we compare the pasting contexts of CaTT to the Batanin trees. Section 6 contains our main result, the comparison of the syntactic category of CaTT and the category of finite computads. In Section 7, we discuss briefly how our result lets us obtain a more direct proof of the equivalence of models of CaTT and algebras of the monad on globular sets induced by the adjunction with computads.

2 The dependent type theory CaTT

2.1 Dependent type theories and categories with families

The dependent type theory CaTT, introduced by Finster and Mimram [15], is a syntactic way to work with weak ω-categories. To introduce a dependent type theory, we consider a fixed set of variables $\mathcal{V}$ that we assume countably infinite, and define mutually inductively the following syntactic objects

- Contexts are lists of pairs $(x : A)$, where $x \in \mathcal{V}$ and A is a type (empty list denoted $\emptyset$)
- Substitutions are lists of $\langle x \mapsto t\rangle$ where $x \in \mathcal{V}$ and t is a term (empty list denoted $\langle\rangle$)
- Types are expressions built from type constructors
- Terms are expressions either of the form $\mathrm{var}\, x$ where x is a variable, or built from term constructors

We note that only some of those syntactic objects will be considered valid in the type theory, as explained below.

A substitution γ may act on a type A (resp. on a term t), producing a type $A[\gamma]$ (resp. a term $t[\gamma]$). The definition may vary depending on the type theory, but we require that for every variable x, the term $(\mathrm{var}\, x)[\gamma]$ is the last term associated to x in γ if it exists, and $\mathrm{var}\, x$ otherwise. A set of variables is

$$\frac{}{\emptyset \vdash}\ (\textsc{ec}) \qquad \frac{(x : A) \in \Gamma}{\Gamma \vdash \mathrm{var}\, x : A}\ (\textsc{var}) \qquad \frac{\Gamma \vdash}{\Gamma \vdash \langle\rangle : \emptyset}\ (\textsc{es})$$

$$\frac{\Gamma \vdash A \qquad x \notin \mathrm{Var}(\Gamma)}{(\Gamma, x : A) \vdash}\ (\textsc{cc}) \qquad \frac{\Delta \vdash \gamma : \Gamma \qquad (\Gamma, x : A) \vdash \qquad \Delta \vdash t : A[\gamma]}{\Delta \vdash \langle \gamma, x \mapsto t\rangle : (\Gamma, x : A)}\ (\textsc{sc})$$

Fig. 1. Context and substitution rules

associated to contexts, terms, types and substitutions, denoted Var, and we always define

$$\mathrm{Var}(\emptyset) = \emptyset \qquad \mathrm{Var}(\Gamma, x : A) = \mathrm{Var}(\Gamma) \cup \{x\}$$
$$\mathrm{Var}(\langle\rangle) = \emptyset \qquad \mathrm{Var}(\langle\gamma, x \mapsto t\rangle) = \mathrm{Var}(\gamma) \cup \mathrm{Var}(t)$$
$$\mathrm{Var}(\mathrm{var}\, x) = \{x\}$$

Additionally, contexts, substitutions, types and terms are subjects to well-formedness conditions that are expressed in the form of the following judgements, presented here along with their intended meaning.

$\Gamma \vdash$	Γ is a valid context	$\Gamma \vdash A$	A is a valid type in Γ
$\Delta \vdash \gamma : \Gamma$	γ is a valid substitution from Δ to Γ	$\Gamma \vdash t : A$	t is a type of type A in Γ

We also require all dependent type theories to satisfy the context and substitutions forming rules, as well as the variable forming rules presented in Fig. 1. Additionally, in the theories that we consider, the action of substitution on terms defines composition of substitutions which is associative and unital.

This lets us consider the *syntactic category* $\mathcal{S}_{\mathbb{T}}$ associated to a dependent type theory $\mathbb{T}$, whose objects are the valid contexts and whose morphisms $\Delta \to \Gamma$ are the substitutions $\Delta \vdash \gamma : \Gamma$. the syntactic category carries a structure of category with families, in the sense of Dybjer [14]. We recall that the category Fam of *families of sets* is the category with objects set-indexed families of sets $(A_i)_{i\in I}$ and morphisms are pairs $(\phi, f_i) \colon (A_i)_{i\in I} \to (B_j)_{j\in J}$ of a function $\phi \colon I \to J$ together with a family of functions $f_i \colon A_i \to B_{\phi(i)}$ for every $i \in I$. A *category with families* $\mathcal{C}$ is a category with a chosen terminal object $\mathbf{1}$, equipped with a Fam-valued presheaf $T \colon \mathcal{C}^{\mathrm{op}} \to \mathrm{Fam}$, denoted by

$$T(\Gamma) = (\mathrm{Tm}(\Gamma; A))_{A\in \mathrm{Ty}\,\Gamma}.$$

Additionally, there exists an extension operation associating to every pair of an object $\Gamma \in \mathcal{C}$ and $A \in \mathrm{Ty}\,\Gamma$, an object $(\Gamma.A)$ together with a projection $\pi_{\Gamma,A} : (\Gamma.A) \to \Gamma$ and an element $v_{\Gamma,A} \in \mathrm{Tm}(\Gamma; \mathrm{Ty}(\pi_{\Gamma,A})A)$ satisfying the following universal property: For every object Δ with a map $\gamma : \Delta \to \Gamma$ and an element $u \in \mathrm{Tm}(\Delta; \mathrm{Ty}(\gamma)A)$, there exists a unique map $\langle\gamma, u\rangle : \Delta \to (\Gamma.A)$ such that

$$\pi_{\Gamma,A}\langle\gamma, u\rangle = \gamma \qquad \mathrm{Tm}(\langle\gamma, u\rangle)(v_{\Gamma,A}) = u.$$

A morphism of categories with families $F \colon \mathcal{C} \to \mathcal{D}$ consists of a functor $\mathcal{C} \to \mathcal{D}$ preserving the chosen terminal object, together with a natural transformation $F^T : T^{\mathcal{C}} \Rightarrow T^{\mathcal{D}} \circ F$ such that the extension operation is preserved on the nose. By definition of the category of families, we see that F^T amounts to functions

$$F^{\mathrm{Ty}}_{\Gamma} \colon \mathrm{Ty}^{\mathcal{C}}(\Gamma) \to \mathrm{Ty}^{\mathcal{D}}(F\Gamma) \qquad F^{\mathrm{Tm}}_{\Gamma,A} \colon \mathrm{Tm}^{\mathcal{C}}(\Gamma; A) \to \mathrm{Ty}^{\mathcal{D}}(F\Gamma; F^{\mathrm{Ty}}_{\Gamma} A)$$

satisfying the expected naturality conditions.

The structure of category with families for the syntactic category of a dependent type theory is given by choosing $\mathrm{Ty}(\Gamma)$ to be the valid types in context Γ, choosing $\mathrm{Tm}(\Gamma; A)$ to be the set of terms of type A in context Γ, and the extension operation to be the context extension of the theory. The universal projection $\pi_{\Gamma,A} : (\Gamma.A) \to \Gamma$ is the obvious weakening and the universal element $v_{\Gamma,A}$ is the last variable of the context. We have formalised [6] the category with families structure of the syntactic category for the two dependent type theories `GSeTT` and `CaTT` that we consider here. For the sake of readability, we introduce the dependent type theories using named variables, however we consider the contexts in the syntactic category to be quotiented by α-renaming, which can be achieved in practice by presenting the type theory with De Bruijn indices for instance [12]. We make sure to present the theories in a way that makes translating to De Bruijn indices trivial.

$$*[\gamma] = * \qquad (t \to_A u)[\gamma] = t[\gamma] \to_{A[\gamma]} u[\gamma]$$

$$\mathrm{Var}\,(*) = \emptyset \qquad \mathrm{Var}(t \to_A u) = \mathrm{Var}(A) \cup \mathrm{Var}(t) \cup \mathrm{Var}(u)$$

$$\frac{\Gamma \vdash}{\Gamma \vdash *}\ (\text{OBJ}) \qquad \frac{\Gamma \vdash A \qquad \Gamma \vdash t : A \qquad \Gamma \vdash u : A}{\Gamma \vdash t \to_A u}\ (\text{ARR})$$

Fig. 2. Definition of the type theory `GSeTT`

$$\frac{}{(x : *) \vdash_{\mathsf{ps}} x : *}\ (\text{PSS}) \qquad \frac{\Gamma \vdash_{\mathsf{ps}} f : x \to_A y}{\Gamma \vdash_{\mathsf{ps}} y : A}\ (\text{PSD})$$

$$\frac{\Gamma \vdash_{\mathsf{ps}} x : A \qquad y, f \notin \mathrm{Var}(\Gamma)}{\Gamma, y : A, f : x \to_A y \vdash_{\mathsf{ps}} f : x \to_A y}\ (\text{PSE}) \qquad \frac{\Gamma \vdash_{\mathsf{ps}} x : *}{\Gamma \vdash_{\mathsf{ps}}}\ (\text{PS})$$

Fig. 3. Derivation of pasting schemes

$$\mathrm{coh}_{\Gamma,A}[\gamma][\delta] = \mathrm{coh}_{\Gamma,A}[\gamma \circ \delta] \qquad \mathrm{Var}(\mathrm{coh}_{\Gamma,A}[\gamma]) = \mathrm{Var}(\gamma)$$

$$\frac{\Gamma \vdash_{\mathsf{ps}} \qquad \Gamma \vdash u : A \qquad \Gamma \vdash v : A \qquad \Delta \vdash \gamma : \Gamma}{\Delta \vdash \mathrm{coh}_{\Gamma, u \to_A v}[\gamma] : (u \to_A v)[\gamma]}\ (\text{COH})$$

Where Rule (COH) is subject to one of the following side conditions

$$\dim \Gamma \geq 1 \quad \text{and} \quad \begin{cases} \mathrm{Var}(s^{\Gamma}_{\dim \Gamma - 1}) = \mathrm{Var}\, u \cup \mathrm{Var}\, A \\ \mathrm{Var}(t^{\Gamma}_{\dim \Gamma - 1}) = \mathrm{Var}\, v \cup \mathrm{Var}\, A \end{cases} \qquad \text{or} \qquad \begin{cases} \mathrm{Var}(\Gamma) = \mathrm{Var}\, u \cup \mathrm{Var}\, A \\ \mathrm{Var}(\Gamma) = \mathrm{Var}\, v \cup \mathrm{Var}\, A \end{cases}$$

Fig. 4. Definition of the type theory `CaTT`

2.2 The dependent type theories GSeTT and CaTT

We first introduce the dependent type theory `GSeTT`, a dependent type theory describing globular sets. This theory has no term constructors, and two type constructors $*$, taking no argument, and $\to$, taking as a arguments a type A and two terms u and v and producing the type $t \to_A u$. This type theory is completely determined by the definitions and rules given in Fig. 2. The dependent type theory `CaTT` is an extension of `GSeTT`, with one term constructor coh taking as arguments a context Γ, a type A and a substitution γ and producing the term $\mathrm{coh}_{\Gamma,A}[\gamma]$. We note that the brackets here are part of the syntax, and do not denote the application of a substitution. The introduction rule for this term constructor requires the use of the two following auxiliary judgements, subject to the rules presented in Fig. 3

$$\Gamma \vdash_{\mathsf{ps}} \quad \Gamma \text{ is a pasting scheme} \qquad \Gamma \vdash_{\mathsf{ps}} x : A \quad \text{auxiliary judgement}$$

Moreover, we need the following auxiliary definitions. The *dimension* of a type and a context are defined recursively by

$$\dim(*) = -1 \qquad \dim(t \to_A u) = \dim A + 1$$
$$\dim(\emptyset) = -1 \qquad \dim(\Gamma, x : A) = \max(\dim(\Gamma), \dim A + 1)$$

For every pasting context $\Gamma \vdash_{\mathsf{ps}}$ and every $k \in \mathbb{N}$, we define two contexts $\partial_k^{\pm}(\Gamma)$ of `GSeTT` by induction on the derivation, where rm is the function removing the last element of a list and $\epsilon \in \{+, -\}$,

$$\partial_k^{\epsilon}((x : *) \vdash_{\mathsf{ps}} x : *) = (x : *) \tag{PSS}$$

$$\partial_k^{\epsilon}(\Gamma, y : A, f : x \to_A y \vdash_{\mathsf{ps}} fx \to_A y) = \begin{cases} \partial_k^{\epsilon}(\Gamma \vdash_{\mathsf{ps}} x : A) & \dim A > k \\ (\partial_k^{\epsilon}(\Gamma \vdash_{\mathsf{ps}} x : A), y : A, f : x \to_A y) & \dim A \leq k - 1 \\ \partial_k^{-}(\Gamma \vdash_{\mathsf{ps}} x : A) & \dim A = k, \epsilon = - \\ (\mathrm{rm}\, \partial_k^{+}(\Gamma \vdash_{\mathsf{ps}} x : A), y : A) & \dim A = k, \epsilon = + \end{cases} \tag{PSE}$$

$$\partial_k^{\epsilon}(\Gamma \vdash_{\mathsf{ps}} y : A) = \partial_k^{\epsilon}(\Gamma \vdash_{\mathsf{ps}} f : x \to_A y) \tag{PSD}$$

$$\partial_k^{\epsilon}(\Gamma \vdash_{\mathsf{ps}}) = \partial_k^{\epsilon}(\Gamma \vdash_{\mathsf{ps}} x : *) \tag{PS}$$

The first author proved [6] that $\partial_k^{\pm}\Gamma$ are actually pasting contexts. By weakening, we get two substitutions $\Gamma \vdash s_k^{\Gamma} : \partial_k^{-}\Gamma$ and $\Gamma \vdash t_k^{\Gamma} : \partial_k^{+}\Gamma$ that assign each variable to the variable with the same name. Fig. 4 then completely defines the dependent type theory `CaTT`. The definition given here differs slightly from the original definition due to Finster and Mimram [15]. We unite the two rules they present into a single one and we use an alternative, but equivalent, second side condition. The equivalence has been shown by the first author [5, Section 3.5.1]. We note that the two side conditions can be further unified into a single one

$$\mathrm{Var}(s^{\Gamma}_{\dim A}) = \mathrm{Var}(u) \cup \mathrm{Var}(A) \qquad \mathrm{Var}(t^{\Gamma}_{\dim A}) = \mathrm{Var}(v) \cup \mathrm{Var}(A)$$

using that $s_k^{\Gamma} = t_k^{\Gamma} = \mathrm{id}$ for $k \geq \dim \Gamma$ and a result of the first author [5, Lemma 76]. Moreover, we observe that the side condition can only be used when $\dim \Gamma \leq \dim A + 1$.

3 Computads for weak ω-categories

Computads are combinatorial structures for presenting higher categories, first invented by Street [27], and later generalised by Burroni [11] and Batanin [3]. They consist of sets of generators with specified source and target cells, which are cells of the higher category freely generated by the computad. The apparent circularity of this description of computads is resolved by defining the category of computads and free higher categories on computads mutually recursively.

Following Dean, Finster, Markakis, Reutter and Vicary [13], to define computads for ω-categories, we need to start by defining globular sets and Batanin trees. A *globular set* X consists of a set X_n for every $n \in \mathbb{N}$ together with source and target functions $\mathrm{src}, \mathrm{tgt} \colon X_{n+1} \to X_n$ subject to the globularity conditions

$$\mathrm{src} \circ \mathrm{src} = \mathrm{src} \circ \mathrm{tgt} \qquad \mathrm{tgt} \circ \mathrm{src} = \mathrm{tgt} \circ \mathrm{tgt}$$

We call elements of X_n the *n-cells* of X, and we call src and tgt the *source* and *target* functions. Morphisms of globular sets $f \colon X \to Y$ are sequences of functions $f_n \colon X_n \to Y_n$ commuting with the source and target functions. Globular sets with their morphisms form a category gSet, which is a presheaf topos. We denote by $\mathbb{D}^n$ the globular set represented by $n \in \mathbb{N}$, and we denote by $\mathbb{S}^{n-1}$ the globular set obtained by removing the unique n-cell of $\mathbb{D}^n$. We will also denote the obvious inclusion by $\iota_n \colon \mathbb{S}^{n-1} \to \mathbb{D}^n$.

Batanin trees are a family of globular sets, corresponding to *pasting schemes*, that play the role of arities for the operations of ω-categories. The set of Batanin trees Bat is defined inductively by one constructor

$$\mathrm{br} \colon \mathrm{List}\,\mathrm{Bat} \to \mathrm{Bat}\,.$$

In particular, there exists a Batanin tree br[] with no branches. The *dimension* of a Batanin tree is the height of the corresponding tree, and it is defined recursively by

$$\dim(\mathrm{br}[]) = 0 \qquad \dim(\mathrm{br}[B_1, \dots, B_n]) = \max(\dim B_1 + 1, \dim(\mathrm{br}[B_2, \dots, B_n])).$$

The *k-boundary* of a Batanin tree B is the Batanin tree defined recursively by

$$\partial_0 B = \mathrm{br}[] \qquad \partial_{k+1} \mathrm{br}[B_1, \dots, B_n] = \mathrm{br}[\partial_k B_1, \dots, \partial_k B_n].$$

A globular set $\mathrm{Pos}\, B$ can be assigned to each Batanin tree B, whose cells we call *positions*. This globular set can be generated inductively, as shown by Dean et al. [13]. Equivalently, it can be described recursively by the formula

$$\mathrm{Pos}(\mathrm{br}[B_1, \dots, B_n]) = \bigvee_{i=1}^{n} \Sigma\, \mathrm{Pos}(B_i).$$

as shown in our previous work [8]. Here the suspension operation Σ sends a globular set X to the bipointed globular set with two 0-cells v_- and v_+ and globular set of cells from v_- to v_+ given by X, and the wedge sum of two bipointed globular sets $X \vee Y$ is obtained by the coproduct of X and Y by identifying the second basepoint of X with the first basepoint of Y. Positions of a Batanin tree are precisely the sectors of the corresponding planar tree, as explained by Berger [9]. We illustrate this correspondence in Fig. 6. The positions of the k-boundary of B may be included back in the positions of B via the source and target inclusions

$$s_k^B, t_k^B \colon \mathrm{Pos}(\partial_k B) \to \mathrm{Pos}(B)$$

which are similarly defined recursively by s_0^B and t_0^B being the morphisms out of the 0-disk picking the left and right basepoint of $\mathrm{Pos}(B)$ and

$$s_{k+1}^{\mathrm{br}[B_1,\dots,B_n]} = \bigvee_{i=1}^{n} \Sigma s_k^{B_i} \qquad t_{k+1}^{\mathrm{br}[B_1,\dots,B_n]} = \bigvee_{i=1}^{n} \Sigma t_k^{B_i}.$$

We note that the source and target inclusions, satisfy equations dual to the globularity conditions, and that they are identities for $k \geq \dim B$.

The category of n-computads Comp_n is then defined inductively on $n \in \mathbb{N}$ together with four functors

$$\begin{aligned} \mathrm{Free}_n &\colon \mathrm{gSet} \to \mathrm{Comp}_n & \mathrm{Cell}_n &\colon \mathrm{Comp}_n \to \mathrm{Set} \\ \mathrm{tr}_{n-1} &\colon \mathrm{Comp}_n \to \mathrm{Comp}_{n-1} & \mathrm{Sphere}_n &\colon \mathrm{Comp}_n \to \mathrm{Set}\,. \end{aligned}$$

The functor Free_n views a globular set as an n-computad. The functor tr_{n-1} forgets the top-dimensional generators of a computad. The functors Cell_n and Sphere_n send an n-computad to the set of n-cells of the ω-category it presents, and the set of pairs of parallel n-cells respectively. In the same mutual induction, three natural transformations are defined

$$\mathrm{pr}_1 \colon \mathrm{Sphere}_n \Rightarrow \mathrm{Cell}_n \qquad \mathrm{pr}_2 \colon \mathrm{Sphere}_n \Rightarrow \mathrm{Cell}_n \qquad \mathrm{bdry} \colon \mathrm{Cell}_n \Rightarrow \mathrm{Sphere}_{n-1} \mathrm{tr}_{n-1}$$

where pr_i are the obvious projections and bdry takes a cell to its source and target. Finally, for every Batanin tree B, a subset of spheres $\mathrm{Full}_n(B) \subseteq \mathrm{Sphere}_n(\mathrm{Free}_n \mathrm{Pos}(B))$ is defined to be *full*.

The induction starts by letting Comp_{-1} be the terminal category and Sphere_{-1} the functor picking the terminal set. For $n \in \mathbb{N}$, an n-computad C consists of an $(n-1)$-computad C_{n-1}, a set of generators V_n^C, and an attaching function $\phi_n^C \colon V_n^C \to \mathrm{Sphere}_{n-1}(C_{n-1})$. A morphism $f \colon C \to D$ of n-computads consists of a morphism $f_{n-1} \colon C_{n-1} \to D_{n-1}$ and a function $f_V \colon V_n^C \to \mathrm{Cell}_n(D)$ such that

$$\mathrm{bdry}_{n,D} \circ f_{n,V} = \mathrm{Sphere}_{n-1}(f_{n-1}) \circ \phi_n^C.$$

The functor tr_{n-1} is the obvious first projection.

The set of n-cells of a computad C is defined inductively by two constructors. The first constructor var produces an n-cell out of a generator $v \in V_n^C$. The second constructor coh is only used when $n > 0$ and produces an n-cell out of a Batanin tree B of dimension at most n, a full $(n-1)$-sphere A of B and a morphism $f\colon \mathrm{Free}_n \mathrm{Pos}(B) \to C$. The boundaries of those cells are given recursively by

$$\mathrm{bdry}_{n,C}(\mathrm{var}\, v) = \phi_n^C(v) \qquad \mathrm{bdry}_{n,C}(\mathrm{coh}(B, A, f)) = \mathrm{Sphere}_{n-1}(f_{n-1})(A)$$

An n-sphere of an n-computad C is then a pair of n-cells (a, b) with the same boundary, and the projection natural transformations are the obvious ones. Composition of morphisms and the action of a morphism on n-cells are given mutually recursively by

$$\mathrm{Cell}_n(g)(\mathrm{var}\, v) = g_V(v)$$
$$\mathrm{Cell}_n(g)(\mathrm{coh}(B, A, f)) = \mathrm{coh}(B, A, g \circ f)$$
$$g \circ f = (g_{n-1} \circ f_{n-1}, \mathrm{Cell}_n(g) \circ f_{n,V})$$

The n-computad $\mathrm{Free}_n X$ consists of $\mathrm{Free}_{n-1} X$, the set X_n of n-cells and the attaching function

$$\phi_n^{\mathrm{Free}\, X}(x) = (\mathrm{var}\,\mathrm{src}\, x, \mathrm{var}\,\mathrm{tgt}\, x).$$

The fullness condition for a sphere $(a, b) \in \mathrm{Sphere}_n(\mathrm{Free}_n \mathrm{Pos}(B))$ is a condition on the *supports* of the cells a and b, which are the sets of n-positions used in the definition of a and b respectively. More precisely, the support of a cell $c \in \mathrm{Cell}_n(C)$ is the set of n-dimensional generators of C defined by

$$\mathrm{supp}_n(\mathrm{var}\, v) = \{v\} \qquad \mathrm{supp}_n(\mathrm{coh}(B, A, f)) = \bigcup_{p \in \mathrm{Pos}_n(B)} \mathrm{supp}_n(f_V(p)).$$

A sphere (a, b) is then declared to be full when the supports of a and b contain precisely the n-positions in the image of the source and target inclusions s_n^B and t_n^B respectively, and the sphere $\mathrm{bdry}_n(a)$ is full.

Finally, the category of computads Comp is the limit of the categories of n-computads over the obvious truncation functors tr_n. We will denote the projections out of the limit by $\mathrm{Tr}_n\colon \mathrm{Comp} \to \mathrm{Comp}_n$. By the universal property of the limit, the functors Free_n give rise to a functor

$$\mathrm{Free}\colon \mathrm{gSet} \to \mathrm{Comp},$$

while the functor $\mathrm{Cell}_n \mathrm{Tr}_n$ together with the natural transformations $\mathrm{pr}_i \,\mathrm{bdry}_n$ give rise to a functor in the opposite direction

$$\mathrm{Cell}\colon \mathrm{Comp} \to \mathrm{gSet}\,.$$

It is shown by Dean et al. [13] that the functor Free is left adjoint to Cell; the unit being the morphism of globular sets given by var and the counit being the morphism of computads viewing cells of C as generators of $\mathrm{Free}\,\mathrm{Cell}\, C$. The algebras of the monad T induced by this adjunction are precisely the ω-categories of Leinster [20].

3.1 The category with families structures on gSet *and* Comp

The opposite of the category of globular sets $\mathrm{gSet}^{\mathrm{op}}$ can be equipped with the structure of a category with families. The presheaf of families $T^{\mathrm{gSet}}\colon \mathrm{gSet} \to \mathrm{Fam}$ sends an object X to the family

$$\mathrm{Ty}^{\mathrm{gSet}}(X) = \coprod_{n \in \mathbb{N}} \mathrm{gSet}(\mathbb{S}^{n-1}, X) \qquad \mathrm{Tm}^{\mathrm{gSet}}(X; (n, A)) = \{x\colon \mathbb{D}^n \to X \mid x \circ \iota_n = A\}$$

By the Yoneda lemma and the definition of $\mathbb{S}^n$, we see that types of a globular set X are in bijection to pairs of cells of X with common source and target; terms of a type (n, A) are n-cells of X whose source and target are given by A. The extension of a globular set X by a type (n, A) is given by the following pushout square in gSet,

$$\begin{array}{ccc} \mathbb{S}^{n-1} & \xhookrightarrow{\iota_n} & \mathbb{D}^n \\ {\scriptstyle A}\downarrow & \ulcorner & \downarrow{\scriptstyle v_{X,A}} \\ X & \xrightarrow[\pi_{X,A}]{} & X.A \end{array}$$

or more concretely by adjoining a new cell of dimension n to X whose source and target are determined by A. The universal property of the extension $X.A$ in the category with families structure is exactly the universal property of the pushout in the category gSet.

The opposite of the category of computads $\mathrm{Comp}^{\mathrm{op}}$ can similarly equipped with the structure of a category with families by sending a computad C to the family

$$\mathrm{Ty}^{\mathrm{Comp}}(C) = \coprod_{n \in \mathbb{N}} \mathrm{Comp}(\mathrm{Free}\,\mathbb{S}^{n-1}, C) \qquad \mathrm{Tm}^{\mathrm{Comp}}(C; (n, A)) = \{c \colon \mathrm{Free}\,\mathbb{D}^n \to C \mid c \circ \mathrm{Free}\,\iota_n = A\}$$

By representability of Sphere_n and Cell_n [13, Corollary 4.2], we have that the set of types of a computad C is the union of the sets of n-spheres of C over all n, and that the set of terms of a type (n, A) is the set of cells with boundary sphere A. The extension of a computad C by a type (n, A) is given similarly by the following pushout square in Comp.

$$\begin{array}{ccc} \mathrm{Free}\,\mathbb{S}^{n} & \xhookrightarrow{\mathrm{Free}\,\iota_{n+1}} & \mathrm{Free}\,\mathbb{D}^{n+1} \\ {\scriptstyle A}\downarrow & \ulcorner & \downarrow{\scriptstyle v_{C,A}} \\ C & \xrightarrow[\pi_{C,A}]{} & C.A \end{array}$$

which exists by [13, Proposition 6.3]; it is obtained by adjoining a new generator of dimension n to C with boundary sphere A. By construction, the functor $\mathrm{Free} \colon \mathrm{gSet} \to \mathrm{Comp}$ preserves the chosen pushout square, so the functor $\mathrm{Free}^{\mathrm{op}} \colon \mathrm{gSet}^{\mathrm{op}} \to \mathrm{Comp}^{\mathrm{op}}$ is part of a morphism of categories with families, equipped with the natural transformations given by the action of Free on morphisms.

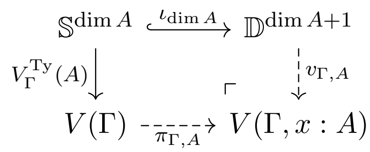
4 Globular sets and GSeTT

The type theory `GSeTT` has been studied in detail by Benjamin, Finster and Mimram [7] where it was shown that its category of Set-models is equivalent to the category of globular sets, and that its syntactic category is equivalent to the opposite of the subcategory of finite globular sets. Here by finite globular sets, we mean globular sets X such that $\amalg_{n \in \mathbb{N}} X_n$ is finite. More explicitly, using that contexts, types, terms and substitutions of `GSeTT` have unique derivations, we can define a morphism of categories with families $V \colon \mathcal{S}_{\mathtt{GSeTT}} \to \mathrm{gSet}^{\mathrm{op}}$ recursively on the syntax by the rules given in Fig. 5. Functoriality and naturality of the assignment V are straightforward to check by mutual induction, while preservation of the extension operation is immediate from the definition.

Since the type theory `GSeTT` has no term constructors, the cells of the globular set $V\Gamma$ are precisely the valid terms of the context Γ, that is, its variables. It follows that the functor constructed here is isomorphic to the functor introduced by Benjamin, Finster and Mimram [7, Definition 15], hence fully faithful with essential image the finite globular sets. Fullness and faithfulness amount to morphisms of globular sets $V\Gamma \to V\Delta$ being functions from the variables of Δ to the variables of Γ compatible with the source and target functions. Essential surjectivity onto finite globular sets amounts to every finite globular set X being a finite extension $\emptyset.A_1.A_2.\cdots.A_n$ for some finite sequence of spheres $A_k \in \mathrm{Ty}^{\mathrm{gSet}}(\emptyset.A_1.A_2.\cdots.A_{k-1})$. For similar reasons, the natural transformations V^{Ty} and V^{Tm} are invertible.

$$V(\emptyset) = \emptyset \qquad V_{\Gamma,\emptyset}(\langle\rangle) = \langle\rangle \qquad V_{\Gamma}^{\mathrm{Ty}}(*) = \langle\rangle$$

$$V_{\Delta,(\Gamma,x:A)}(\langle\gamma,t\rangle) = \langle V_{\Delta,\Gamma}(\gamma), V_{\Delta,A[\gamma]}^{\mathrm{Tm}}(t)\rangle$$



$$V_{\Gamma}^{\mathrm{Ty}}(u \to_A v) = \langle V_{\Gamma,A}^{\mathrm{Tm}}(u), V_{\Gamma,A}^{\mathrm{Tm}}(v)\rangle$$

$$V_{(\Gamma,x:A),B}^{\mathrm{Tm}}(\mathrm{var}\, v) = \begin{cases} v_{\Gamma,A} & \text{if } v = x \\ \pi_{\Gamma,A} \circ V_{\Gamma,B}^{\mathrm{Tm}}(\mathrm{var}\, v) & \text{otherwise} \end{cases}$$

Fig. 5. The morphism $V : \mathcal{S}_{\mathtt{GSeTT}} \to \mathrm{gSet}^{\mathrm{op}}$

5 Pasting contexts and Batanin trees

Before comparing the type theory `CaTT` with the category of computads, we need to compare the arities of the operation coh in the two theories, namely the set of pasting contexts PsCtx and the set of Batanin trees Bat. To compare the two notions, we will use the auxiliary notions of *smooth zigzag sequence* and Street's *globular cardinal* [28], and their boundaries. The correspondence between the four notions is illustrated in Fig. 6 for a particular globular cardinal.

Following Weber [31, Section 4], we may assign functorially to every globular set X, a preorder $\mathrm{Sol}(X)$, whose carrier is the set $\amalg_{n\in\mathbb{N}} X_n$ of cells of X and whose relation $\blacktriangleleft$ is the reflexive and transitive closure of the relation $\prec$ defined by $\mathrm{src}\, x \prec x$ and $x \prec \mathrm{tgt}\, x$ for every cell x. A *globular cardinal* is a globular set X for which $\mathrm{Sol}\, X$ is a non-empty finite total order. We define further the k-boundary of a globular cardinal X to be the globular cardinal $\partial_k X$ whose n-cells are given by

$$(\partial_k X)_n = \begin{cases} X_n & \text{if } n < k \\ (X_k)_{/\sim} & \text{if } n = k \\ \emptyset & \text{if } n > k \end{cases}$$

where two k-cells are related by $\sim$ if and only if they have the same source and target. The source and target inclusions $s_k^X, t_k^X : \partial_k X \to X$ are the morphisms of globular sets that are identity on cells of dimension less than k, and they pick the least and greatest representative of each equivalence class of k-cells respectively. It is easy to see that the k-boundary operation is functorial and the source and target inclusions are natural with respect to isomorphisms. The notion of boundary we define here is a mild generalisation of the boundary of a globular cardinal as defined by Benjamin, Finster and Mimram [7], where they define the source and target $\partial^{\pm} X$ of a globular cardinal instead. The precise relation between the two notions is that

$$\partial^+ X = \mathrm{im}(s_{\dim X - 1}^X) \qquad\qquad \partial^- X = \mathrm{im}(t_{\dim X - 1}^X).$$

where $\dim X$ is the maximum of the dimension of the cells of X.

Lemma 5.1 *The suspension ΣX of a globular cardinal X is a globular cardinal whose basepoints are its minimum and maximum cell. Moreover, the following equalities hold:*

$$\partial_{k+1}(\Sigma X) = \Sigma(\partial_k X) \qquad\qquad s_{k+1}^{\Sigma X} = \Sigma s_k^X \qquad\qquad t_{k+1}^{\Sigma X} = \Sigma t_k^X.$$

Proof. The total order $\mathrm{Sol}(\Sigma X)$ is obtained from the order $\mathrm{Sol}\, X$ by freely adjoining a minimum and maximum element. The equalities follow easily from the definition of the suspension. ◻

Lemma 5.2 *The wedge sum $X \vee Y$ of two cardinals X and Y whose basepoints are their minimum and maximum cells is again a globular cardinal with the same property. Moreover, there exists an isomorphism, making the following two triangles commute:*

$$\begin{array}{ccc} \partial_{k+1}(X \vee Y) & \xrightarrow{\sim} & (\partial_{k+1}X) \vee (\partial_{k+1}Y) \\ & {\scriptstyle s^{X\vee Y}_{k+1}} \searrow \quad \swarrow {\scriptstyle s^X_{k+1} \vee s^Y_{k+1}} & \\ & X \vee Y & \end{array} \qquad \begin{array}{ccc} \partial_{k+1}(X \vee Y) & \xrightarrow{\sim} & (\partial_{k+1}X) \vee (\partial_{k+1}Y) \\ & {\scriptstyle t^{X\vee Y}_{k+1}} \searrow \quad \swarrow {\scriptstyle t^X_{k+1} \vee t^Y_{k+1}} & \\ & X \vee Y & \end{array}$$

Proof. The order $\mathrm{Sol}(X \vee Y)$ is obtained from by identifying the maximum cell of X with the minimum cell of Y. The isomorphism is the identity on cells of dimension at most k, and it is given on $(k+1)$-cells by the isomorphism of sets

$$(X_{k+1} \amalg Y_{k+1})_{/\sim} \xrightarrow{\sim} (X_{k+1})_{/\sim} \amalg (Y_{k+1})_{/\sim}$$

which exists, since no element of X_{k+1} is related by $\sim$ to an element of Y_{k+1}. Commutativity of the triangles follows easily by the definition of the source and target inclusions. □

Proposition 5.3 *The globular set $\mathrm{Pos}(B)$ is a globular cardinal for every Batanin tree B. Moreover, there exists an isomorphism $\partial_k \mathrm{Pos}(B) \cong \mathrm{Pos}(\partial_k B)$ making the following diagrams commute*

$$\begin{array}{ccc} \partial_k \mathrm{Pos}(B) & \xrightarrow{\sim} & \mathrm{Pos}(\partial_k B) \\ & {\scriptstyle s^{\mathrm{Pos}(B)}_k} \searrow \quad \swarrow {\scriptstyle s^B_k} & \\ & \mathrm{Pos}(B) & \end{array} \qquad \begin{array}{ccc} \partial_k \mathrm{Pos}(B) & \xrightarrow{\sim} & \mathrm{Pos}(\partial_k B) \\ & {\scriptstyle t^{\mathrm{Pos}(B)}_k} \searrow \quad \swarrow {\scriptstyle t^B_k} & \\ & \mathrm{Pos}(B) & \end{array}$$

Proof. This is an immediate consequence of Lemmas 5.1 and 5.2, and the fact that the unit of the wedge sum $\mathbb{D}^0$ is also a globular cardinal. □

We now show every globular cardinal is isomorphic to one of the form $\mathrm{Pos}(B)$ for unique B. To do so, we use the *smooth zigzag sequences* introduced by Weber [31]. A smooth exact sequence is a non-empty finite sequence of natural numbers $(m_1, \dots, m_n)$ such that $m_1 = m_n = 0$ and $|m_i - m_{i-1}| = 1$ for all $i < n$. To each globular cardinal X, we may assign a smooth zigzag sequence $\mathrm{Zig}\, X = (\dim x_1, \dots, \dim x_k)$, where $x_1 \prec \dots \prec x_k$ are the cells of X in increasing order. Conversely, from a smooth zigzag sequence $(m_1, \dots, m_n)$, we may construct a globular cardinal $\mathrm{Card}(m_1, \dots, m_n)$, whose k-cells are indices $i \leq n$ such that $m_i = k$ and whose source and target functions are given by

$$\mathrm{src}(i) = \max\{j < i \mid m_j = m_i - 1\} \qquad \mathrm{tgt}(i) = \min\{j > j \mid m_j = m_i - 1\}$$

Isomorphic globular cardinals are sent to the same smooth zigzag sequence, and

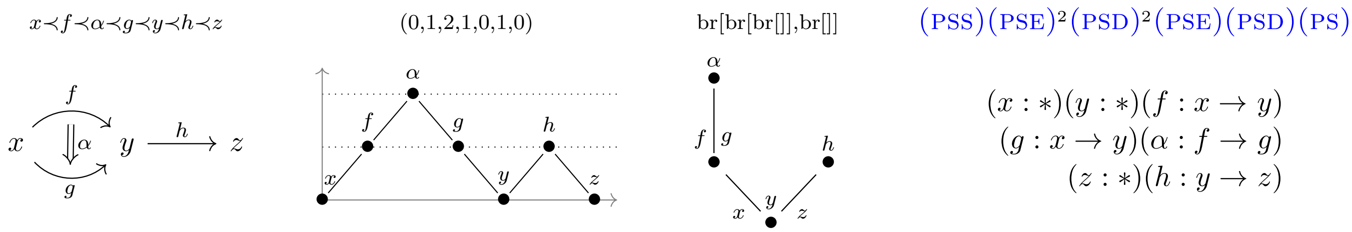
$$\mathrm{Zig} \circ \mathrm{Card}(m_\bullet) = m_\bullet \qquad \mathrm{Card} \circ \mathrm{Zig}(X) \cong X$$

so that smooth zigzag sequences are in bijection with isomorphism classes of globular cardinals.

To show that Pos is a bijection onto isomorphism classes of globular cardinals, it suffices to show that $\mathrm{Zig} \circ \mathrm{Pos}$ is a bijection onto smooth zigzag sequences. From the proofs of Proposition 5.3, we can derive a recursive formula for the composite $\mathrm{Zig} \circ \mathrm{Pos}$:

$$\mathrm{Zig}\,\mathrm{Pos}(\mathrm{br}[B_1, \dots, B_n]) = \mathop{\#}_{i=1}^{n} (\mathrm{Zig}\,\mathrm{Pos}(B_1))^+$$

where $(m_1, \dots, m_n)^+ = (0, m_1 + 1, \dots, m_n + 1, 0)$ and $\#$ is the operation concatenating two smooth zigzag sequences by identifying the last element of the first with the first element of the second. The inverse of



(a) A globular cardinal (b) A smooth zigzag sequence (c) A Batanin tree (d) A pasting context

Fig. 6. The correspondence between globular cardinals, smooth zigzag sequences, Batanin trees and pasting contexts.

Zig ∘ Pos is given by the function Tree defined recursively on the length of a sequence by

$$\mathrm{Tree}(0, m_1^1, \dots, m_{n_1}^1, 0, \dots, 0, m_1^l, \dots, m_{n_l}^l, 0) = \mathrm{br}[B_1, \dots, B_l]$$
$$B_i = \mathrm{Tree}(m_1^i - 1, \dots, m_{n_i}^i - 1)$$

where $m_i^j > 0$ for all indices i, j. Hence, we have proven the following:

Proposition 5.4 *For each globular cardinal X, there exists unique Batanin tree B such that $\mathrm{Pos}(B) \cong X$.*

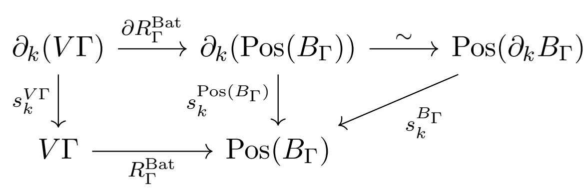
The last step in comparing pasting contexts to Batanin trees is to show that the former are also in bijection to isomorphism classes of globular cardinals. This has already been shown by Benjamin, Finster and Mimram. The globular set $V\Gamma$ associated to a pasting context Γ, seen as a context of GSeTT, is a globular cardinal [7, Proposition 40] and conversely every globular cardinal is isomorphic to $V\Gamma$ for a unique pasting context Γ [7, Proposition 42]. Intuitively, the order $\mathrm{Sol}(V\Gamma)$ is the order in which the variables appear in the right hand side of the auxiliary judgements in the derivation tree of $\Gamma \vdash_{\mathsf{ps}}$. Conversely, to each smooth zigzag sequence $(m_0, \dots, m_k)$, we may assign a derivation of the form $\Gamma \vdash_{\mathsf{ps}}$ by starting from the rule (PSS), iteratively using the rule (PSE) when $m_i > m_{i-1}$ and the rule (PSD) otherwise, until the sequence is exhausted, and concluding with the rule (PS). Since the derivation of the judgement $\Gamma \vdash_{\mathsf{ps}}$ is unique, one can check that $\mathrm{Zig} \circ V$ is a bijection with inverse the assignment described above. The following Theorem summarises the results of this section.

Theorem 5.5 *There exists a bijection $B_\bullet \colon \mathrm{PsCtx} \to \mathrm{Bat}$ together with a family of isomorphisms*

$$R_\Gamma^{\mathrm{Bat}} \colon V\Gamma \xrightarrow{\sim} \mathrm{Pos}(B_\Gamma)$$

sending the image of $V(s_{\dim\Gamma-1}^{\Gamma})$ to the image of $s_{\dim\Gamma-1}^{B_\Gamma}$, and similarly for the target inclusions.

Proof. The bijection $B_\bullet$ is the composite $\mathrm{Tree} \circ \mathrm{Zig} \circ V$. By Proposition 5.3 and naturality, the following diagram commutes:



Hence, the isomorphism R_Γ^{Bat} identifies the image of $s_k^{V\Gamma}$ with that of $s_k^{B_\Gamma}$. The former for $k = \dim\Gamma - 1$ has been shown to be equal to the image of $V(s_k^\Gamma)$ by Benjamin, Finster and Mimram [7, Lemma 45]. ☐

6 Computads and CaTT

We are now in the position to compare the type theory CaTT with the category of computads, extending the functor V of Section 4 by using the bijection $B_\bullet$ of Section 5. We define a morphism of categories with

$$R(\emptyset) = \emptyset \qquad R_{\Gamma,\emptyset}(\langle\rangle) = \langle\rangle \qquad R_{\Delta,(\Gamma,x:A)}(\langle\gamma, t\rangle) = \langle R_{\Delta,\Gamma}(\gamma), R^{\mathrm{Tm}}_{\Delta,A[\gamma]}(t)\rangle$$

$$\begin{array}{ccc} \mathrm{Free}\,\mathbb{S}^{\dim A} & \xrightarrow{\mathrm{Free}\,\iota_{\dim A}} & \mathrm{Free}\,\mathbb{D}^{\dim A+1} \\ {\scriptstyle R^{\mathrm{Ty}}_{\Gamma}(A)}\downarrow & \ulcorner & \downarrow{\scriptstyle \upsilon_{\Gamma,A}} \\ R(\Gamma) & \xrightarrow[\pi_{\Gamma,A}]{} & R(\Gamma, x : A) \end{array} \qquad \begin{array}{c} R^{\mathrm{Ty}}_{\Gamma}(*) = \langle\rangle \\ \\ R^{\mathrm{Ty}}_{\Gamma}(u \to_A v) = \langle R^{\mathrm{Tm}}_{\Gamma,A}(u), R^{\mathrm{Tm}}_{\Gamma,A}(v)\rangle \end{array}$$

$$R^{\mathrm{Tm}}_{(\Gamma,x:A),B}(\mathrm{var}\, v) = \begin{cases} \upsilon_{\Gamma,A} & \text{if } v = x \\ \pi_{\Gamma,A} \circ R^{\mathrm{Tm}}_{\Gamma,B}(\mathrm{var}\, v) & \text{otherwise} \end{cases}$$

$$R^{\mathrm{Tm}}_{\Delta,A}(\mathrm{coh}_{\Gamma,A}[\gamma]) = \mathrm{coh}(B_\Gamma, \mathrm{Free}\, R^{\mathrm{Bat}}_{\Gamma} \circ R^{\mathrm{Ty}}_{\Gamma} A, R_{\Gamma,\Delta}(\gamma) \circ (\mathrm{Free}\, R^{\mathrm{Bat}}_{\Gamma})^{-1})$$

Fig. 7. The morphism $R\colon \mathcal{S}_{\texttt{CaTT}} \to \mathrm{Comp}^{\mathrm{op}}$

families $R\colon \mathcal{S}_{\texttt{CaTT}} \to \mathrm{Comp}^{\mathrm{op}}$ by induction on the syntax, as indicated in Fig. 7. Mutually inductively, we verify that the following diagram commutes

$$\begin{array}{ccc} \mathcal{S}_{\texttt{GSeTT}} & \xrightarrow{V} & \mathrm{gSet}^{\mathrm{op}} \\ {\scriptstyle J}\big\downarrow & & \big\downarrow{\scriptstyle \mathrm{Free}} \\ \mathcal{S}_{\texttt{CaTT}} & \xrightarrow[R]{} & \mathrm{Comp}^{\mathrm{op}} \end{array}$$

where the functor J views contexts and substitutions of `GSeTT` as contexts and substitutions of `CaTT`. For the last case in Fig. 7, we implicitly use the fact that morphisms $\mathbb{S}^n \to C$ are in natural bijection with elements of $\mathrm{Sphere}_n(C)$. Moreover, for this case to be well-defined, we need to show that for a type A satisfying the side condition of the rule (COH), the sphere $(\mathrm{Free}\, R^{\mathrm{Bat}}_{\Gamma} \circ R^{\mathrm{Ty}} A)$ is again full. Commutativity of the square is immediate by Free preserving strictly the chosen pushouts of sphere inclusions. To prove fullness, we see from the inductive presentation that we may canonically identify variables of Γ with the generators of $R\Gamma$. Under this identification, it suffices to prove that

$$\mathrm{supp}(R^{\mathrm{Tm}}_{\Gamma,A} u) = \mathrm{Var}(u) \cup \mathrm{Var}(A)$$

Indeed, for a type $A' = u \to_A v$ satisfying the side condition, we have by the support lemma [13, Lemma 7.3] and Theorem 5.5, we see that

$$\begin{aligned} \mathrm{supp}(\mathrm{Free}\, R^{\mathrm{Bat}}_{\Gamma} \circ R^{\mathrm{Tm}}_{\Gamma,A}(u)) &= R^{\mathrm{Bat}}_{\Gamma}(\mathrm{supp}\, R^{\mathrm{Tm}}_{\Gamma,A}(u)) = R^{\mathrm{Bat}}_{\Gamma}(\mathrm{Var}(u) \cup \mathrm{Var}(A)) \\ &= R^{\mathrm{Bat}}_{\Gamma}(\mathrm{Var}(s^{\Gamma}_{\dim A'})) = \mathrm{im}(s^{B_\Gamma}_{\dim A'}) \end{aligned}$$

and similarly for the support of the target. The equation on the support of a term follows again immediately by induction from the definitions of R^{Tm}, Var and the support. This concludes the definition of the morphism R. Functoriality of R and naturality of R^{Ty} and R^{Tm} can be shown by a further mutual induction, using the universal property of the pushout defining R.

Theorem 6.1 *The functor $R\colon \mathcal{S}_{\texttt{CaTT}} \to \mathrm{Comp}^{\mathrm{op}}$ is fully faithful with essential image the finite computads. The natural transformations R^{Ty} and R^{Tm} are invertible.*

Proof. We first prove faithfulness together with injectivity of the natural transformations R^{Ty} and R^{Tm} by mutual induction on the syntax. Suppose that $R^{\mathrm{Tm}}(u) = R^{\mathrm{Tm}}(v)$ for terms $\Gamma \vdash u : A$ and $\Gamma \vdash v : A$.

Then if u is a variable, so is v and in that case $u = v$. If $u = \mathrm{coh}_{\Gamma,A}[\gamma]$ is a coherence term, then so is $v = \mathrm{coh}_{\Gamma',A'}[\delta]$ and the above equality gives:

$$\begin{aligned} B_\Gamma &= B_{\Gamma'} \\ \mathrm{Free}(R^{\mathrm{Bat}}_\Gamma) \circ R^{\mathrm{Ty}}(A) &= \mathrm{Free}(R^{\mathrm{Bat}}_{\Gamma'}) \circ R^{\mathrm{Ty}}(A') \\ R\gamma \circ \mathrm{Free}(R^{\mathrm{Bat}}_\Gamma)^{-1} &= R\gamma' \circ \mathrm{Free}(R^{\mathrm{Bat}}_{\Gamma'})^{-1}. \end{aligned}$$

Using the injectivity of the map $B_\bullet$ and the inductive hypothesis for $R^{\mathrm{Ty}}(A)$ and $R\gamma$, we conclude that $u = v$. Similarly, one can prove that if $R^{\mathrm{Ty}}(A) = R^{\mathrm{Ty}}(A')$, then $A = A'$ by case analysis on A. Finally, one can prove that if $R\gamma = R\delta$, then $\gamma = \delta$ by case analysis on γ and the uniqueness part of the universal property of the pushout defining R on contexts.

We then show fullness together with surjectivity of R^{Ty} and R^{Tm} again by mutual induction on dimension, and structural induction on the cells and morphisms of computads. Fix a context $\Delta \vdash$. First note that R^{Tm} is surjective on generator cells since it induces a bijection between the variables of Δ and the generators of $R\Delta$. Consider an n-cell of the form $c = \mathrm{coh}(B, A, f)$. By Theorem 5.5, there exists a pasting context Γ such that $B = B_\Gamma$. By surjectivity of R^{Ty} on $(n-1)$-spheres, and by structural induction, there exist a type $\Gamma \vdash A'$, and a substitution $\Delta \vdash \gamma : \Gamma$ respectively, satisfying

$$(\mathrm{Free}\, R^{\mathrm{Bat}}_\Gamma)^{-1} \circ A = R^{\mathrm{Ty}}(A') \qquad\qquad f \circ (\mathrm{Free}\, R^{\mathrm{Bat}}_\Gamma) = R\gamma.$$

By the equality of support proven in the construction of R, the type A' must satisfy the side-condition of the rule (COH). By construction, $c = R^{\mathrm{Tm}}(\mathrm{coh}_{\Gamma,A'}[\gamma])$, hence R^{Tm} is surjective on n-cells. Surjectivity of R^{Ty} onto n-spheres follows by surjectivity of R^{Tm} on n-cells. Finally, consider a context Γ, and a morphism $f : R\Gamma \to R\Delta$. If Γ is the empty context, then f is the unique morphism out of the initial object, which is the image of the empty substitution. If $\Gamma = (\Gamma', x : A)$ is an extension, then f decomposes as $f = \langle g, c\rangle$ with $g : R\Gamma' \to R\Delta$ and c a cell of $R\Gamma$. By structural induction, $g = R\gamma$ and $c = R^{\mathrm{Tm}}(u)$ for some substitution $\Delta \vdash \gamma : \Gamma'$ and some term u in Γ. By definition of R on morphisms, $f = R\langle \gamma, u\rangle$.

Finally, we show that the essential image of R coincides with the subcategory of finite computads. By construction, the computad $R\Gamma$ is finite with exactly as many generators as the length of Γ. We prove the converse by induction on the number of generators. If C is a computad with no generators, then $C = \emptyset = R\emptyset$. Otherwise, we choose a generator $v \in V^C_n$ of maximal dimension and consider the computad C' obtained by removing that generator from C. By construction, there exists a pushout square

$$\begin{array}{ccc} \mathrm{Free}\,\mathbb{S}^{n-1} & \xrightarrow{\mathrm{Free}\,\iota_n} & \mathrm{Free}\,\mathbb{D}^n \\ {\scriptstyle \phi^C_n(v)}\downarrow & & \downarrow{\scriptstyle v} \\ C' & \xrightarrow[\iota]{} & C \end{array}$$

By the inductive hypothesis, there exists some context Γ together with an isomorphism $f \colon C' \xrightarrow{\sim} R\Gamma$. Then by uniqueness of the pushout, we have that

$$C \cong R(\Gamma, x : A),$$

where $A = (R^{\mathrm{Ty}}_\Gamma)^{-1}(f \circ \phi^C_n(v))$. This concludes the proof that R is essentially surjective onto finite computads. □

Corollary 6.2 *The syntactic category $\mathcal{S}_{\mathtt{CaTT}}$ is equivalent to the opposite of the full subcategory of* Comp *consisting of finite computads.*

7 Models of CaTT

We conclude our study with a a brief discussion about the models of the dependent type theory `CaTT`. Denote PsCtx the full subcategory of $\mathcal{S}_{\texttt{CaTT}}$ whose objects are the contexts satisfying $\Gamma \vdash_{\texttt{ps}}$. Explicitly, PsCtx is the category whose objects are pasting contexts, and morphisms are the substitutions in `CaTT` between them, viewing them as contexts in `CaTT`. Benjamin, Finster and Mimram [7] have showed that the models of the dependent theory `CaTT` are equivalent to the copresheaves over PsCtx that preserve a class of limits, called globular products.

On the other hand, the second author's thesis [25] shows that the free ω-category monad T has arities the class of globular pasting diagrams, which means that the nerve functor

$$N_T \colon \mathrm{Alg}_T \to [\mathrm{Bat}^{\mathrm{op}}, \mathrm{Set}]$$
$$N_T(X)(B) = \mathrm{Alg}_T(F\,\mathrm{Pos}\,B, X)$$

is fully faithful, where Bat is the category of Batanin trees and morphisms of computads, and F is the free T-algebra functor. It is essentially surjective onto presheaves that preserve a classs of colimits, called globular sums. Alternatively, this follows from the comparison theorem of Dean et al. [13, Corollary 7.36], and the work of Berger [9, Theorem 1.17].

The functor R we have defined in Section 6 satisfies the following equality for every pasting context Γ,

$$R\Gamma = \mathrm{Free}\,V\Gamma = \mathrm{Free}\,\mathrm{Pos}\,B_\Gamma.$$

Since $B_\bullet$ is bijective and the functor R is fully faithful, the latter restricts to an isomorphism of categories $\mathrm{PsCtx} \cong \mathrm{Bat}^{\mathrm{op}}$. One can check that under this isomorphism globular products correspond to globular sums. Composing those equivalences, we conclude that the categories of T-algebras and models of `CaTT` are equivalent.

References

[1] Altenkirch, T. and O. Rypacek, *A syntactical approach to weak omega-groupoids*, in: *Computer Science Logic (CSL'12) - 26th International Workshop/21st Annual Conference of the EACSL (CSL 2012)*, page 15 pages, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik GmbH, Wadern/Saarbruecken, Germany (2012).
https://doi.org/10.4230/LIPICS.CSL.2012.16

[2] Ara, D., *Sur les ∞-groupoïdes de Grothendieck et une variante ∞-catégorique*, Thèse de doctorat, Université Paris Diderot (Paris 7) (2010).
https://www.i2m.univ-amu.fr/perso/dimitri.ara/files/these.pdf

[3] Batanin, M. A., *Computads for finitary monads on globular sets*, in: E. Getzler and M. Kapranov, editors, *Higher Category Theory*, volume 230 of *Contemporary Mathematics*, pages 37–57, American Mathematical Society, Providence, Rhode Island (1998), ISBN 978-0-8218-1056-9 978-0-8218-7821-7.
https://doi.org/10.1090/conm/230/03337

[4] Batanin, M. A., *Monoidal globular categories as a natural environment for the theory of weak n-categories*, Advances in Mathematics **136**, pages 39–103 (1998), ISSN 00018708.
https://doi.org/10.1006/aima.1998.1724

[5] Benjamin, T., *A type theoretic approach to weak ω-categories and related higher structures*, Thèse de doctorat, Institut Polytechnique de Paris (2020).
https://hal.science/tel-03106197

[6] Benjamin, T., *Formalisation of Dependent Type Theory: The Example of CaTT*, in: *27th International Conference on Types for Proofs and Programs (TYPES 2021)* (2022).
https://doi.org/10.4230/LIPICS.TYPES.2021.2

[7] Benjamin, T., E. Finster and S. Mimram, *Globular weak ω-categories as models of a type theory*, Higher Structures (2021).
https://doi.org/10.48550/ARXIV.2106.04475

[8] Benjamin, T. and I. Markakis, *Opposites of weak ω-categories and the suspension and hom adjunction* (2024). https://doi.org/10.48550/ARXIV.2402.01611

[9] Berger, C., *A cellular nerve for higher categories*, Advances in Mathematics **169**, pages 118–175 (2002), ISSN 00018708. https://doi.org/10.1006/aima.2001.2056

[10] Bourke, J., *Iterated algebraic injectivity and the faithfulness conjecture*, High.Struct. **4**, pages 183–210 (2020). https://doi.org/10.21136/HS.2020.13

[11] Burroni, A., *Higher-dimensional word problems with applications to equational logic*, Theoretical Computer Science **115**, pages 43–62 (1993), ISSN 03043975. https://doi.org/10.1016/0304-3975(93)90054-W

[12] De Bruijn, N., *Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the Church-Rosser theorem*, Indagationes Mathematicae (Proceedings) **75**, pages 381–392 (1972), ISSN 13857258. https://doi.org/10.1016/1385-7258(72)90034-0

[13] Dean, C. J., E. Finster, I. Markakis, D. Reutter and J. Vicary, *Computads for weak ω-categories as an inductive type* **450**, page 109739, ISSN 00018708. https://doi.org/10.1016/j.aim.2024.109739

[14] Dybjer, P., *Internal type theory*, in: G. Goos, J. Hartmanis, J. Leeuwen, S. Berardi and M. Coppo, editors, *Types for Proofs and Programs*, volume 1158, pages 120–134, Springer Berlin Heidelberg, Berlin, Heidelberg (1996), ISBN 978-3-540-61780-8 978-3-540-70722-6. https://doi.org/10.1007/3-540-61780-9_66

[15] Finster, E. and S. Mimram, *A type-theoretical definition of weak ω-categories*, in: *Proceedings of the 32nd Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 1–12, ACM, Reyjavik Iceland (2017), ISBN 978-1-5090-3018-7. https://doi.org/10.5555/3329995.3330059

[16] Garner, R., *Homomorphisms of higher categories*, Advances in Mathematics **224**, pages 2269–2311 (2010), ISSN 00018708. https://doi.org/10.1016/j.aim.2010.01.022

[17] Grothendieck, A., *Pursuing stacks* (1983). https://doi.org/10.48550/arXiv.2111.01000

[18] Henry, S. and E. Lanari, *On the homotopy hypothesis in dimension 3*, Theory and Applications of Categories **39**, pages 735–768 (2023). https://doi.org/10.48550/arXiv.1905.05625

[19] Lafont, Y. and F. Métayer, *Polygraphic resolutions and homology of monoids*, Journal of Pure and Applied Algebra **213**, pages 947–968 (2009), ISSN 00224049. https://doi.org/10.1016/j.jpaa.2008.10.005

[20] Leinster, T., *Operads in Higher-Dimensional Category Theory*, Ph.D. thesis, University of Cambrdige (2000). https://doi.org/10.48550/arXiv.math/0011106

[21] Leinster, T., *A Survey of Definitions of n-Category* (2001). https://doi.org/10.48550/arXiv.math/0107188

[22] Leinster, T., *Higher operads, higher categories*, Cambridge University Press, Cambridge, UK (2004), ISBN 978-0-521-53215-0. https://doi.org/10.48550/arXiv.math/0305049

[23] Lumsdaine, P. L., *Weak ω-categories from intensional type theory*, in: P.-L. Curien, editor, *Typed Lambda Calculi and Applications*, volume 5608, pages 172–187, Springer Berlin Heidelberg, Berlin, Heidelberg (2009), ISBN 978-3-642-02272-2 978-3-642-02273-9. https://doi.org/10.1007/978-3-642-02273-9_14

[24] Maltsiniotis, G., *Grothendieck ∞-groupoids, and still another definition of ∞-categories* (2010). https://doi.org/10.48550/arXiv.1009.2331

[25] Markakis, I., *An inductive approach to ω-categories and their computads*, Ph.D. thesis, University of Cambrdige (2024).

[26] Mimram, S., *Towards 3-dimensional rewriting theory*, Log. Meth. Comput. Sci. **10** (2014), ISSN 18605974. https://doi.org/10.2168/LMCS-10(2:1)2014

[27] Street, R., *Limits indexed by category-valued 2-functors*, Journal of Pure and Applied Algebra **8**, pages 149–181 (1976), ISSN 00224049.
https://doi.org/10.1016/0022-4049(76)90013-X

[28] Street, R., *The petit topos of globular sets*, Journal of Pure and Applied Algebra **154**, pages 299–315 (2000), ISSN 00224049.
https://doi.org/10.1016/S0022-4049(99)00183-8

[29] The Univalent Foundations Program, *Homotopy Type Theory: Univalent Foundations of Mathematics* (2013).
https://homotopytypetheory.org/book

[30] van den Berg, B. and R. Garner, *Types are weak ω-groupoids*, Proceedings of the London Mathematical Society **102**, pages 370–394 (2011), ISSN 00246115.
https://doi.org/10.1112/plms/pdq026

[31] Weber, M., *Generic morphisms, parametric representations and weakly cartesian monads.*, TAC **13**, pages 191–234 (2004), ISSN 1201-561X.
https://www.emis.de/journals/TAC/volumes/13/14/13-14.pdf