

UNIVERSITY OF
CAMBRIDGE

MATHEMATICS TRIPOS

Part III

**On Computads for Strict
 ω -Categories**

Easter Term, 2022

Set by

Dr. Jamie Vicary

Essay by

Chiara Sarti

Contents

1	Introduction	2
1.1	Context and Motivation	2
1.2	Overview of Sources	3
1.3	Notational Remarks	4
2	Defining Computads	5
2.1	Graphs, Higher Graphs and Globes	5
2.2	Higher Categories	8
2.3	Pasting Diagrams, Free n -Categories and Computads	9
3	Free (Commutative) Monoids as Computads	13
3.1	Free Monoids and Familiar Representability	13
3.2	The Eckmann-Hilton Argument	15
3.3	Free Commutative Monoids	16
4	Presheaves and Strictness	19
4.1	Shapes and Strictness	19
4.2	Computads by Artin Glueing	19
4.3	2-Computads are Presheaves	21
4.4	3-Computads are not Presheaves	23
4.5	3-Computads are not Cartesian closed	25
5	Conclusion	28

1 Introduction

1.1 Context and Motivation

One of the most remarkable features of category theory is its extensive use of diagrams. Such visual notational devices have proven uniquely apt to illustrate the unifying concepts that have seen to the success of the field. Diagrams allow us to leverage our geometric intuition by abstracting *pastability*: every way we may compose the arrows will yield the same result.

This essay will provide a self-contained discussion illustrating some non-geometric behaviour exhibited by diagrams in higher dimensions, which are formalised by the notion of *computads*. This pathological behaviour is intimately linked with the failure of computads to form a presheaf category. This is ultimately caused by *strictness*, the requirement that cancellation of units or associativity hold “*up to strict equality*”, as opposed to *weakly*, or “*only up to isomorphism*”. In our familiar world of ordinary, 1-dimensional category theory, such constraints are baked into the definition of what it even means to be a category. We will analyse in depth how this innocuous assumption can cause unexpected trouble when passing to higher dimensions. Of course, before delving into the theory would make precise such notions, we owe the reader a good reason for why we may need higher dimensional diagrams at all.

Surprisingly, the answer to this question is, in a sense, baked into the core idea of categorification. If Mac Lane’s development of category theory was motivated by the axiomatisation of Noether’s thesis that “*understanding of a mathematical structure in its proper perspective could be better achieved through a proper understanding of the processes preserving that structure*” [SII12, pg. 4], higher category theory turns this principle onto category theory itself, carrying it to its most complete conclusion. Thus, to understand ordinary 1-categories, we work in a category of (suitably small) categories. In such a 2-category, together with the structure of the individual (1-)categories and functors between them, there is a higher-order structure given by the natural transformations between functors. Our (suitably small) 2-categories, in turn, should be reasoned about within the framework of a 3-category, and so forth towards some notion of an ω -category, where our inductive process terminates.

In the face of our philosophical arguments, the reader might feel deprived of concrete examples. We will move to provide an immediate one:

Example 1.1 ($\mathbb{Z}/2\mathbb{Z}$). Consider the monoid $\mathbb{Z}/2\mathbb{Z}$, presented as

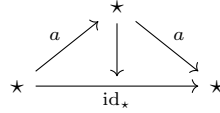
$$\mathbb{Z}/2\mathbb{Z} \cong \langle a | a^2 = e \rangle.$$

Writing $F_{\text{mon}}(-)$ for the free monoid functor, the meaning of this presentation is given by the short exact sequence

$$0 \longrightarrow F_{\text{mon}}(\{r\}) \xrightarrow{r \mapsto a \circ a} F_{\text{mon}}(\{a\}) \longrightarrow \mathbb{Z}/2\mathbb{Z} \longrightarrow 0$$

Recall that, categorically, a monoid is just a category with a single object $\star$ and a suitable set of morphisms. However, making sense of Example 1.1 diagrammatically is not immediately straightforward. According to the usual convention of omitting identities from diagrams, the rightmost morphism should intuitively correspond to the following:

The diagrammatic treatment of relations is less immediate. Morally, exactness should be expressed as the “*arrow between arrows*” in the following diagram



However, a closer inspection of the diagram reveals that instead of an “*arrow between arrows*”, we have an “*arrow between 2-ary and 0-ary composites arrows*”. The need to formally handle such situations rigorously is precisely what led Street to the notion of a *computad*, first defined in [Str76].

The above diagram has another peculiarity worth remarking upon: the same object $\star$ and morphism a are drawn multiple times. Although such behaviour would seem, at first, mostly attributable to the author’s lack of inventiveness, it represents the seeds of the aforementioned non-geometricity of some diagrams. A formal explanation of this point will have to wait until our discussion of the Eckmann-Hilton argument in Section 3.2, but we will already make the link with the notion of strictness for higher categories, which will prevent us from algebraically distinguishing some formally different cells.

Before jumping into our technical discussion, let us briefly place our inquiry within a larger context. The idea of “*relations as higher order generators*”, formalised by free resolutions, can be traced back to Hilbert’s work on invariant theory, a field later subsumed by modern algebraic geometry. His Syzygy Theorem [Hil90], a bound on the length of free resolutions of certain modules, was held by Hilbert as starting point of such a large programme to almost inspire a 24th Problem in his famous list [Thi03].

The century after Hilbert’s address saw the birth and boom of modern computability theory. In rewriting theory, results such as Knuth-Bendix Completion Algorithm [KB83] [GM18, 3.2.1] turned symmetric equalities into directed rule applications. Within it, Hilbert’s “*finitary mathematics programme*” [Thi03] lived its second life, as algebraic finiteness conditions were formally shown to be intimately linked with solubility of word problems [Squ87, SOK94, GM18].

It was exactly studying word problems that Burroni reinvented Street’s computads¹, under the name of polygraphs, as combinatorial devices to keep track of higher order rewrite rules. The need for such higher word problems came from the general situation of what illustrated in Example 1.1: “[...] *many structures of interest are in fact n -categories, while computations in these structures take place in $(n+1)$ -categories*” [Mé03, pg. 148]. Although these themes will largely run parallel to our discussion, we urge the reader to reflect on the implications of our inquiry in this wider context.

1.2 Overview of Sources

The theory of computads has attracted a significant interest over the years. Their status as useful abstract technology has meant that the literature on the subject is peppered with reinventions of the wheel, misattributions, mistakes and a fair share of general confusion. As such, it is worthwhile to have a brief foreword on the sources that have shaped our discussion of the topic.

Most of the definitions reported here are close to those given the discussion of computads by Burroni [Bur93] and Métayer [Mé03, Mé07, Mé16], with the exception of the nomenclature of “*computads*” instead of “*polygraphs*”. These definitions are less apt to

¹Incidentally, we have a rather precise year of birth of the polygraph formalism: a comparative analysis of the ’91 [Bur91] and ’93 [Bur93] versions of Burroni’s paper reveals the addition of a new section, titled “*Polygraphs*”. The ’91 paper prefers the term “*CW presentations of categories*”, which is still present in the ’93 paper, but out of use everywhere else in the literature.

generalisation to weaker notions than the one given by Batanin [Bat98a, Bat02a], but have the advantage of having significantly lower theoretical complexity.

I gained a great deal of conceptual clarity on computads from reading Makkai’s work, in particular from his account given in [Mak05, Mak07]. Although we will not touch on his syntactic approach to computads, it bears mention, alongside Penon’s “*calcul polygraphique*” [Pen99, pg. 61], in light of recent interest in type theoretic treatments of weak higher categories in e.g. [FM17, FRV21].

Also due to Makkai, and Zawadowski, is the first counter-example [MZ01] to the “*presheafness*” claim of [CJ95], which we discuss at length. Although their first account of the counter-example is rather brief, they later followed it up by publishing a more extensive article on it [MZ08], which is lesser known. The direct example of Cartesian closed-ness failing by Cheng, reported in [Che12], has been incredibly illuminating, but beware that it miscites [MZ08] for [MZ01].

Cheng and Leinster’s work on comparisons of different categorical formalisms has been tremendously impactful in shaping my understanding of the current landscape. The interested reader is referred to [Lei01, Lei03, CL04, CG07]. Although the operadic approach to weak higher categories preferred by Leinster in [Lei98, Lei03] is somewhat bureaucratic, it has the benefit of producing really nice visualisations, some of which we adopted. Particularly worthy of mention in Cheng’s visual proof of the Eckmann-Hilton argument given in [CM09], which is reported in our treatment of Theorem 3.7.

Finally, in our discussion of some key results, e.g. Theorems 4.3 and 4.11, we make extensive use of a particular sequent calculus for natural isomorphisms, which bears no recognisable name in the literature. I learnt it from Sterling [SA20], where it appears under the name of “adjoint sequent calculus”, but it has other documented folklore appearances. In attempting to trace its provenance, I stumbled upon the trove of examples in [RR04]. Chasing the references therein, we can trace it through [RZ93] to [Rey91] and finally [LLR89]. These early mentions point to origins in topos-theoretic modalities literature, but its folkloric status is again reinforced by a contemporary appearance in [Wal89], which bears no further references. Hopefully, this brief remark will prevent others from having to undertake the same deep dive through dusty shelves.

1.3 Notational Remarks

We will adopt the following conventions:

- (i) The natural numbers $\mathbb{N}$ include 0,
- (ii) Composites of functions $A \xrightarrow{f} B \xrightarrow{g} C$ will be denoted $g \circ f$,
- (iii) We write $\mathbb{C}, \mathbb{D}$ for categories, $F, G : \mathbb{C} \rightarrow \mathbb{D}$ for functors and $\alpha, \beta : F \rightarrow G$ for natural transformations.
- (iv) We write $A \in \mathbb{C}$ for objects and $f \in \mathbb{C}(A, B)$ for morphisms $f : A \rightarrow B$,
- (v) We write $0 \in \mathbb{C}$ for initial objects, 1 for terminal ones, and $!$ for the unique morphism out of 0 or into 1 ,
- (vi) We write $[\mathbb{C}, \mathbb{D}]$ for functor categories, or $\widehat{\mathbb{C}^{\text{Op}}}$ if $\mathbb{D} = \text{Set}$,
- (vii) Free functors will generally be denoted F and forgetful functors U or V ,
- (viii) We write computads as $\mathcal{C}$ and their set of n -indets as $\mathcal{C}_n$,
- (ix) The category of small n -categories will be denoted Cat_n , with limit Cat_ω ,
- (x) Other categories of relevance will be $\mathbb{G}, \text{Glob}_n, \text{Glob}_\omega, \text{Comp}_n, \text{Comp}_\omega$.

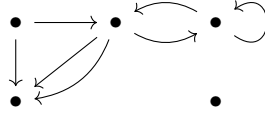
2 Defining Computads

2.1 Graphs, Higher Graphs and Globes

Recall that to every (small) category $\mathbb{C}$ corresponds a graph, with vertices given by the objects of $\mathbb{C}$ and edges by the morphisms. This underlying graph is obtained forgetting the algebraic structure of $\mathbb{C}$. Since our initial aim is to build up some notion of a higher dimensional category, we should focus our initial efforts by fixing a clear notion what a higher graph should morally be.

There are generally two ways this problem is approached, given by the formalism of globular sets $\mathbf{Glob}_\omega$ and of iterated enrichment of graphs $(\mathbf{Grph}_\omega) - \mathbf{Grph}$ ². Some examples are more intuitive when presented in one formalism or another, but they can be shown to be entirely equivalent [Lei03, 1.4.9]. Since most of our examples are more accessible when presented in the globular set formalism, this is the account we will prefer. This decision has also been influence by the fact that, to the author’s knowledge, a definition of computads through iterated enrichment still hasn’t been given in the literature³.

Let us move from the one dimensional case and establish what we should allow to be graphs at all. The following diagram illustrates all the features our notion of graphs needs to include:



In particular, we allow our graphs to contain disconnected vertices, loops and multiple edges between two vertices. Moreover, our edges are directed, so we have a notion of a source s and target map t , assigning to every edge the appropriate vertex. This set of requirements on our notion of graphs sometimes earns them the name of “*quivers*” or “*directed multi-graphs*”, but we’ll just call them “*graphs*”.

We take morphisms of graphs to be functions from vertices to vertices and edges to edges which respect, or commute with, the operations of source and target. Categorically, if we let $\mathbb{G}_1 := (\bullet_1 \rightrightarrows \bullet_0)$ be a category with two objects and two parallel morphisms, then our graphs are nothing but the category of $\mathbf{Set}$ -diagrams on $\mathbb{G}_1^{\text{op}}$, or presheaves $[\mathbb{G}_1^{\text{op}}, \mathbf{Set}]$ on $\mathbb{G}_1$. We’ll denote this presheaf category $\mathbf{Glob}_1$. An object of $\mathbf{Glob}_1$ is a functor F sending $\bullet_0$ to the set of vertices V , $\bullet_1$ to the edges E and the parallel morphisms to the source/target functions.

Clearly, for the purposes of keeping track of generators of free 1-categories, such graphs suffice. Indeed, 1-computads can morally be identified with the graphs presented here. However, as anticipated by in the introductory Example 1.1, the difference between graphs and computads will manifest in higher dimensions.

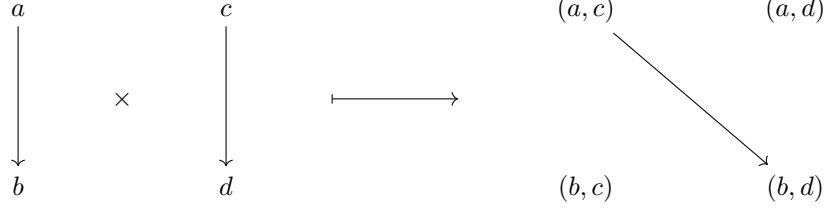
Let us first remark on a couple of examples we can already exact:

Example 2.1 (Non). A “*degenerate graph*”, with empty vertex set and non-empty edge set, is automatically ruled out by our presheaf formalism, as $\emptyset$ is strictly initial in $\mathbf{Set}$. Although this seems more a feature than a bug of our formalism, in the higher dimensional generalisation, it is sometimes desirable to “*skip dimensions*”. We will return to this concept in due course, when discussing free commutative monoids in Section 3.3.

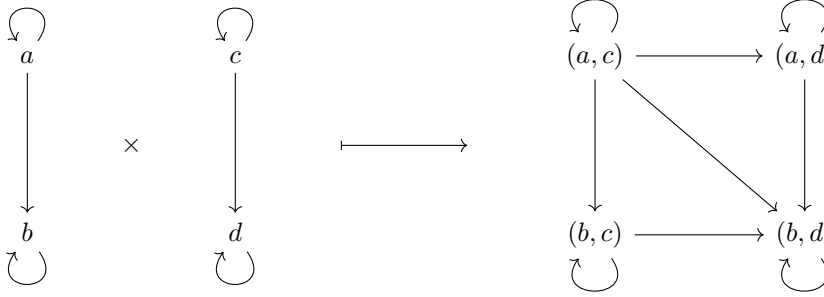
²The reader should be forewarned that some authors in the French literature on computads, such as Burroni [Bur93] and Métayer [Mé03], call $\mathbf{Grph}_\omega$ what we call $\mathbf{Glob}_\omega$.

³For an extensive treatment of enrichment, the interest reader is referred to [Lei00, 3.8].

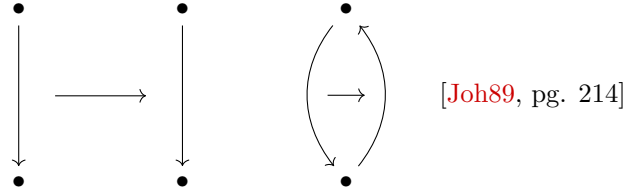
Example 2.2 (Product of Graphs). Since $\mathbf{Glob}_1$ is a presheaf category and $\mathbf{Set}$ is (co)complete, we may take (co)limits of graphs. These can, however, behave in unexpected ways. The following folklore example, which I learnt from Emily Riehl, is particularly illustrative: a binary product of connected graphs can be disconnected!



The culprit here is the lack of “*degenerate identity-like edges*”, which give the intuitive result once added to the picture:



Having fixed a clear notion of graphs as presheaves, the higher dimensional generalisation seems immediate: a higher graph should just be a presheaf on the category $\mathbb{G}_2 := (\bullet_2 \Leftarrow \bullet_1 \Leftarrow \bullet_0)!$ However, such direct extension has the issue of introducing categorically-undesirable “*higher graphs*”, such as



We rule out such pathological cases by adding extra conditions on higher edges:

Definition 2.3 (Globular Sets [Mé07, 2.1]). Let $\mathbb{G}$ be the category with objects the natural numbers and morphisms generated by compositions of $s_n, t_n : n \rightarrow n+1, n \in \mathbb{N}$ subject to the **globularity conditions**:

$$\begin{aligned} s_{n+1} \circ s_n &= t_{n+1} \circ s_n, \\ s_{n+1} \circ t_n &= t_{n+1} \circ t_n. \end{aligned}$$

The category of **globular sets** is the presheaf category

$$\mathbf{Glob}_\omega := [\mathbb{G}^{\text{op}}, \mathbf{Set}].$$

When unambiguous, we will sometimes elide the indices on s_n, t_n and write s and t . Let us provide two immediate examples of prototypical globular sets:

Example 2.4 (n -Globe [Mé07, 2.1]). Let $\mathbf{2} = \{-, +\}$ be a two element set. Since $\mathbf{Glob}_\omega$ is a presheaf category, we may embed any object $n \in \mathbb{G}$ as the globular set

$$\mathbf{Y}(n) : m \mapsto \mathbb{G}(m, n) \cong \begin{cases} \mathbf{2} & \text{if } m < n \\ \mathbf{1} & \text{if } m = n \\ \mathbf{0} & \text{otherwise} \end{cases}$$

We call this globular set the n -**globe**, denoted $\mathbf{O}_n$.

Example 2.5 (ω -Globe). We generalise $\mathbf{O}_n$ to the ω -**globe** $\mathbf{O}$ by taking the globular set which sends every n to $\mathbf{2}$, with constant source and target maps given by

$$\begin{aligned} s_n : x &\mapsto -, \\ t_n : x &\mapsto +. \end{aligned}$$

Towards a third example, we note that for every $n \in \mathbb{N}$, we may functorially truncate $\mathbb{G}$ to a category $\mathbb{G}_n$ with objects the natural numbers $\mathbb{N}_{\leq n}$ bounded by n . We will accordingly denote the category of presheaves on $\mathbb{G}_n$ by $\mathbf{Glob}_n$ and write for the truncation functor $\mathbf{Tr}_n : \mathbf{Glob}_\omega \rightarrow \mathbf{Glob}_n$. Moreover, we may include $\mathbf{l}_n : \mathbf{Glob}_n \rightarrow \mathbf{Glob}_\omega$ by adjoining empty sets as follows:

$$\mathbf{G}(0) \xleftarrow{\quad} \cdots \xleftarrow{\quad} \mathbf{G}(n) \xleftarrow{\quad} \emptyset \xleftarrow{\quad} \cdots$$

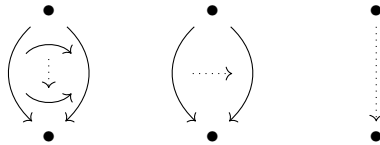
Proposition 2.6 ($\mathbf{l}_n \dashv \mathbf{Tr}_n$). The functors $\mathbf{l}_n \dashv \mathbf{Tr}_n$ are adjoints.

Proof. Let $\mathbf{G} \in \mathbf{Glob}_n, \mathbf{H} \in \mathbf{Glob}_\omega$. We have a natural bijection

$$\frac{(\dots, f_n) : \mathbf{G} \rightarrow \mathbf{Tr}_n(\mathbf{H})}{(\dots, f_n, g_{n+1}, \dots) : \mathbf{l}_n(\mathbf{G}) \rightarrow \mathbf{H}} \exists! g_m : \emptyset \rightarrow \mathbf{H}(m)$$

□

Example 2.7 (n -Sphere [Mé07, 2.1]). Let $\mathbf{O}_n$ be the n -globe constructed in Example 2.4. The n -**sphere** $\partial \mathbf{O}_n$ is the n -truncation $\mathbf{l}_n \mathbf{Tr}_n \mathbf{O}$ of the ω -globe. The following diagram illustrates the action of the truncation functor $\mathbf{Tr}_n$ on $\mathbf{O}$:



The n -globe and n -spheres allow us to define two representable functors $\mathbf{Glob}_\omega \rightarrow \mathbf{Set}$ of particular interest to us:

Definition 2.8 ($\text{Cell}, \text{ParCell}$). For each $n \in \mathbb{G}$ and $G \in \text{Glob}_\omega$, we call an n -**cell** an element of $G(n)$ and an n -**parallel cell** a pair of n -cells with same source and target. By the Yoneda Lemma, the n -**cell functor** is given by

$$\text{Cell}_n : G \mapsto \text{Glob}_\omega(O_n, G) \cong G(n).$$

Similarity, we define the n -**parallel cell functor** by

$$\text{ParCell}_n : G \mapsto \text{Glob}_\omega(\partial O_n, G).$$

2.2 Higher Categories

Having introduced globular sets as our notion of higher graph to arbitrary dimensions, we can equip them with an algebraic structure to obtain our ω -categories. The definition that we report below for strict ω -categories, found in [Lei01, pg. 8], is of heavy combinatorial flavour, but will suffice for our purposes. For any globular set A and integers $n > m \geq 0$, denote $A(n) \times_{A(m)} A(n)$ the pullback of the diagram

$$\begin{array}{ccc} A(n) \times_{A(m)} A(n) & \xrightarrow{\quad} & A(m) \\ \downarrow & \lrcorner & \downarrow t^{n-m} \\ A(n) & \xrightarrow{s^{n-m}} & A(m) \end{array}$$

which is given explicitly by the set

$$A(n) \times_{A(m)} A(n) = \{(x, y) \in A(n) \mid s^{n-m}(x) = t^{n-m}(y)\}.$$

Definition 2.9 (Cat_ω [Lei01, pg. 8]). A (strict) ω -**category** $\mathbb{C}$ is globular set A , together with a family of

- (i) *Compositions* $\circ_m : A(n) \times_{A(m)} A(n) \rightarrow A(n)$,
- (ii) *Identities* $\text{id} : A(n) \rightarrow A(n+1)$,

satisfying

- (i) *Globular conditions for $\circ_m$* : for all $n > m \geq 0$, $(x, y) \in A(n) \times_{A(m)} A(n)$

$$\begin{aligned} s(x \circ_m y) &= s(y), & t(x \circ_m y) &= t(x), & n &= m+1 \\ s(x \circ_m y) &= s(x) \circ_m s(y), & t(x \circ_m y) &= t(x) \circ_m t(y), & n &> m+1. \end{aligned}$$

- (ii) *Globular conditions for id* : for all $n > 0, x \in A(n)$

$$s(\text{id}(x)) = x = t(\text{id}(x)).$$

- (iii) *Unity*: for all $n > m \geq 0, x \in A(n)$

$$\text{id}^{n-m}(t^{n-m}(x)) \circ_m x = x = x \circ \text{id}^{n-m}(s^{n-m}(x)),$$

and for all $x, y \in A(n) \times_{A(m)} A(n)$,

$$\text{id}(x) \circ_m \text{id}(y) = \text{id}(x \circ_m y).$$

(iv) *Associativity*: for all $n > m \geq 0, x, y, z \in \mathbf{A}(n) \times_{\mathbf{A}(m)} \mathbf{A}(n)$

$$(x \circ_m y) \circ_m z = x \circ_m (y \circ_m z).$$

(v) *Interchange*: for all $p > n > m \geq 0, (x, y), (z, w) \in \mathbf{A}(p) \times_{\mathbf{A}(m)} \mathbf{A}(p), (x, z), (y, w) \in \mathbf{A}(n) \times_{\mathbf{A}(m)} \mathbf{A}(n)$

$$(x \circ_n y) \circ_m (z \circ_n w) = (x \circ_m z) \circ_n (y \circ_m w).$$

An ω -**functor** $F : \mathbb{C} \rightarrow \mathbb{D}$ between ω -categories is a morphism of the underlying globular sets which respects compositions and identities.

We also obtain a notion of n -category and n -functor for any $n \in \mathbb{N}$ by truncating the underlying globular sets. We will thus accordingly write $\mathbf{Cat}_\omega$ for the category of all (suitably small) ω -categories and ω -functors, and $\mathbf{Cat}_n$ for its n -dimensional truncation. Clearly, our overarching goal of having the collection of (suitably small) n -categories form an $(n+1)$ -category is satisfied by these notions. Note that the $\mathbf{Cell}_n$ and $\mathbf{ParCell}_n$ functors, introduced in Definition 2.8, carry over to ω -categories.

Let us bring this long definition to reality through a familiar example:

Example 2.10 (Chain Complexes [Mé03, 2.3]). To every chain complex $(A_\bullet, \partial)$ of abelian groups corresponds a strict ω -category $\mathbb{A}$, constructed as follows. We take each set of n -cells to be

$$\mathbf{Cell}_n(\mathbb{A}) = \prod_{i=0}^n A_i,$$

with source, target maps given by

$$\begin{aligned} s : (a_0, \dots, a_n, a_{n+1}) &\mapsto (a_0, \dots, a_n), \\ t : (a_0, \dots, a_n, a_{n+1}) &\mapsto (a_0, \dots, a_n + \partial a_{n+1}). \end{aligned}$$

Since $\partial^2 = 0$, the globular conditions are satisfied and thus we have constructed a globular set. It remains to specify the identities and compositions:

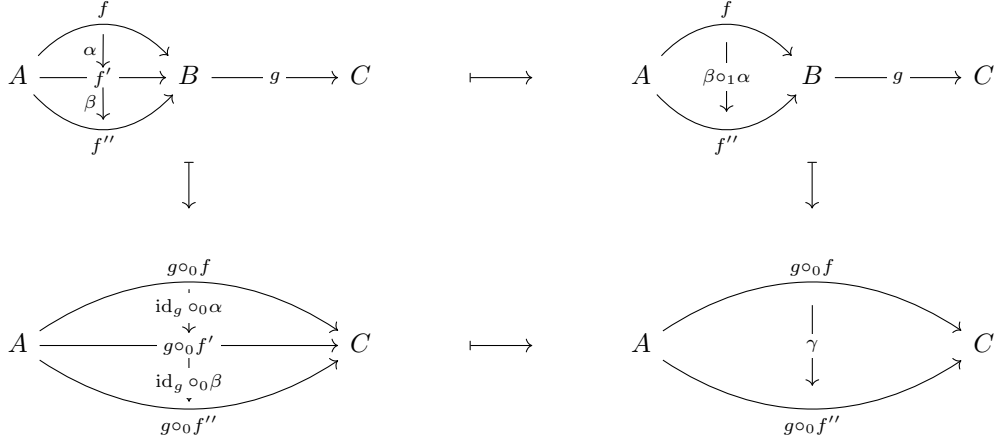
$$\begin{aligned} \text{id} : (a_0, \dots, a_n) &\mapsto (a_0, \dots, a_n, 0), \\ \circ_m : ((a_0, \dots, a_n), (b_0, \dots, b_n)) &\mapsto (a_0, \dots, a_m, a_{m+1} + b_{m+1}, \dots, a_n + b_n). \end{aligned}$$

Checking that this satisfies the strict ω -category axioms amounts to a few tedious but straightforward computations, which we omit.

2.3 Pasting Diagrams, Free n -Categories and Computads

Having introduced a notion of ω -categories, our goal will now be to provide a suitable notion of pastability of higher diagrams, out of which we will naturally arrive at a definition of computads. Generalising from the one dimensional case, it may seem that globular that a morphism of a globular set $G \rightarrow \mathbf{U}(\mathbb{C})$ into the underlying globular set of a strict ω -category $\mathbb{C}$ should be the correct notion for a pasting diagram. Indeed, the axioms of a strict ω -category ensure that every way to compose the 2-cells α and β in

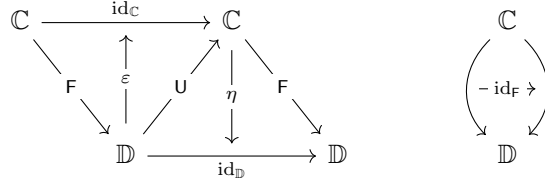
the pasting diagram



will yield the same 2-cell γ , and moreover will do so up to strict equality. The typical formalisation of pasting diagrams is as graded sets, a formalism popularised by Johnson's [Joh89], which offers a precise proof of the pasting theorem for ω -categories. An alternative perspective, due to Batanin [Bat98b] and used extensively by Leinster in [Lei98, Lei00], is to view pasting diagrams as trees. A proof of equivalence of these formalisms is given in [Lei00, 4.2.1].

Coming back to our proposed definition of pasting diagrams as morphism of globular sets, we see that there are, unfortunately, key examples that this notion cannot capture. We present one close to the spirit to the introductory Example 1.1:

Example 2.11 (Adjunctions [Joh89, (1)]). Let $\mathbb{C}, \mathbb{D}$ be (1-)categories. Two functors $F : \mathbb{C} \rightarrow \mathbb{D}, U : \mathbb{D} \rightarrow \mathbb{C}$ being adjoint is a 2-categorical property specified in terms of equality of pasting diagrams representing triangular equations, one of which is



The leftmost diagram cannot be realised as a morphism of globular sets, as the source of the 2-cell ϵ , as well as the target of η , are not 1-cells, but formal composites of 1-cells.

Intuitively, to capture such diagrams, the source and target maps of a globular set, at each dimension n , must be allowed to have codomain in the free $(n-1)$ -category over the set of $(n-1)$ -cells. Diagrammatically, it means moving away from the picture

$$A(0) \longleftarrow A(1) \longleftarrow A(2) \longleftarrow \dots$$

to the picture found in [Bur93, fig. 6]

$$\begin{array}{ccccccc}
 A(0) & & A(1) & & A(2) & & \dots \\
 \downarrow \text{id} & \swarrow & \downarrow \text{id} & \swarrow & \downarrow \text{id} & \swarrow & \\
 A(0)^* & \longleftarrow & A(1)^* & \longleftarrow & A(2)^* & \longleftarrow & \dots
 \end{array} \quad (\dagger)$$

writing X^* for the free n -category on X . This was understood by Street and reported in [Str76], which contains the first definition of computads, restricted to the 2-dimensional case. Our account will follow [Mé03, 3.1] in reporting the generalised to arbitrary dimensions of n -computads and free n -categories. We will proceed by simultaneous induction on dimension n , with the induction step will be given by the following key construction. Consider the pullback diagram in $\mathbf{Cat}_\omega$, which can be shown to be (co)complete by standard arguments [Lei03, F.2.2], given by

$$\begin{array}{ccccc}
 \mathbf{Cat}_{n+1} & & & & \\
 \downarrow \text{Tr}_n & \swarrow \text{V}_{n+1} & & \searrow \text{U} & \\
 & \mathbf{Cat}_n^+ & \xrightarrow{\pi_2} & \mathbf{Glob}_{n+1} & \\
 & \downarrow \pi_1 & \lrcorner & \downarrow \text{Tr}_n & \\
 & \mathbf{Cat}_n & \xrightarrow{\text{U}} & \mathbf{Glob}_n &
 \end{array}$$

Note that, to simplify notation, we identified each $\mathbf{Cat}_n$ with its image in $\mathbf{Cat}_\omega$, and similarly for $\mathbf{Glob}_n$ and $\mathbf{Glob}_\omega$. Explicitly, the objects of $\mathbf{Cat}_n^+$ are pairs $(\mathbb{C}, \mathbf{A})$ where $\mathbb{C}$ is an n -category and $\mathbf{A}$ is an $(n+1)$ -globular set which agrees with the globular set $\mathbf{U}(\mathbb{C})$ up to dimension n . Thus, the only new data in $\mathbf{A}$ is the generating set $(n+1)$ -cells of the next “free level” of $\mathbb{C}$.

Definition 2.12 (n -Indeterminates). Let $(\mathbb{C}, \mathbf{A}) \in \mathbf{Cat}_n^+$. We call $\mathbf{A}(n+1)$ the set of $(n+1)$ -indeterminates, or $(n+1)$ -indets.

Thus, given an $(n+1)$ -category, we may forget the $(n+1)$ -dimensional categorical structure, and obtain through $\mathbf{V}_{n+1}$ an n -category and a set of $(n+1)$ -indeterminates. Our freely generated $(n+1)$ -category construction will have to proceed in the other direction. We thus avail ourselves of the following result, whose proof we omit:

Proposition 2.13 ($\mathbf{F}_{n+1} \dashv \mathbf{V}_{n+1}$). The functor $\mathbf{V}_{n+1}$, given from the universal property of pullbacks, has a left adjoint $\mathbf{F}_{n+1}$ such that $\pi_1 \mathbf{F}_{n+1} \mathbf{V}_{n+1} : \mathbf{Cat}_{n+1} \rightarrow \mathbf{Cat}_n$ is the truncation functor Tr_n .

Proof. Standard argument. A much more general proof is given in [Bat98a, §3]. $\square$

Definition 2.14 (Freely Generated n -Category). A **freely generated** $(n+1)$ -category $\mathbb{C}$ is a category isomorphic, as an $(n+1)$ -category, to $\mathbf{F}_{n+1}(\mathbb{C}, \mathbf{A})$ for some $(\mathbb{C}, \mathbf{A}) \in \mathbf{Cat}_n^+$.

A general cell of a free n -category will not be given by an n -indet, but by a formal composite of n -indets. The choice of terminology reflects the clear analogy with free generation of algebraic structures. For example, the ring of polynomials in n variables $R[X_1, \dots, X_n]$ is given by freely-adjointing the set of indeterminates $\{X_1, \dots, X_n\}$ to R [Mak05, pg. 6]. Indeed, a universal property for freely generated ω -categories can be given in a similar fashion to the one for groups or rings. This for example done in [Str87, §4] or [Joh89, pg. 224], but the diagrams are significantly more involved because of the presence of source and target maps in the picture. An alternative view is instead taken by Guetta’s [Gue21], which defines free strict ω -categories in terms of a suitable notion of “basis” [Gue21, 3.5].

With this machinery in place, we can finally make formal sense of Example 2.11 and the successive Diagram (†). Although it is not enough to just specify an element of $\mathbf{Cat}_n^+$ as our pasting diagrams, as it would only keep track of the indeterminates for one dimension, we may take a family of compatible objects in each $\mathbf{Cat}_i^+$ for $i \leq n$. We proceed inductively:

Construction 2.15 (n -Computads [Mé03, 3.1]). A **0-computad** $\mathcal{C}$ is a set, and the **free 0-category** $F_0(\mathcal{C})$ on it is the discrete category on the set.

Assume now that we have a construction for an n -computad $\mathcal{C}$ and the free n -category $F_n(\mathcal{C})$ on it. An $(n+1)$ -**computad** $\mathcal{D}$ is an n -computad $\mathcal{C}$ together with an object $(F_n(\mathcal{C}), A) \in \mathbf{Cat}_n^+$, i.e. a choice of $(n+1)$ -indets, together with maps $s, t : A(n+1) \rightarrow U_n(F_n(\mathcal{C}))$. Finally, the **free** $(n+1)$ -**category** $F_{n+1}(\mathcal{D})$ on it is $F_{n+1}(F_n(\mathcal{C}), A)$.

It is easy to see that this construction gives exactly the desired Diagram (†). The only missing data is the inclusion of each set of indets into their respective free category, but that is easily recovered from the unit of the adjunction $\mu_{n+1} : \text{id}_{\mathbf{Cat}_n^+} \rightarrow V_{n+1}F_{n+1}$. Writing $\mathcal{C}_n$ for the set of n -indets of the m -computad $\mathcal{C}$, for $n \leq m$, we can immediately define a category of computads and their infinite-dimensional generalisations:

Definition 2.16 ($\mathbf{Comp}_n, \mathbf{Comp}_\omega$). A morphism of n -computads $f : \mathcal{C} \rightarrow \mathcal{D}$ is a family of function $(f_i : \mathcal{C}_i \rightarrow \mathcal{D}_i)$ on the indeterminates for each $i \leq n$, making all the relevant diagrams commute. We write $\mathbf{Comp}_n$ for the category of n -computads and morphisms of n -computads, and $\mathbf{Comp}_\omega$ for their limit.

Formally, an ω -computad is a sequence of compatible n -computads for every n , which we can think of as in Diagram (†) “*with no limitation to the right*” [Bur93, pg. 50]. This uses once again completeness of $\mathbf{Cat}_\omega$, which is established for example in [Lei03, F.2.2]. The discussion therein is interesting in its own right, as it contains an explicit description of the free strict ω -category monad on $\mathbf{Glob}_\omega$.

Having covered a lot of ground, from globular sets to free higher categories and computads, the next Section will introduce several key examples of computads.

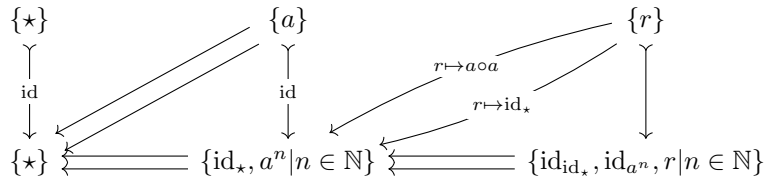
3 Free (Commutative) Monoids as Computads

3.1 Free Monoids and Familiar Representability

Having concluded our previous Section with the moderately abstract definition of computads, our immediate aim will now turn to gaining intuition about them through a few examples. These will naturally lead to the notion of “*familiar representability*”, which will be key for the rest of our discussion. We will then move to defy our freshly-gained intuition on computads by introducing the main theoretical result for this Section, the Eckmann-Hilton argument, as Theorem 3.7. This will allow us to finally show an example of some baffling non-geometric behaviour of high dimensional diagrams, as promised all the way back in the introduction.

Let us start by returning to our introductory Example 1.1 and give a satisfactory answer as to how to diagrammatically interpret our presentation of $\mathbb{Z}/2\mathbb{Z}$.

Example 3.1 ($\mathbb{Z}/2\mathbb{Z}$ [Mé03, 3.2]). $\mathbb{Z}/2\mathbb{Z}$ can be represented by a computad with a 0-indet $\star$, a 1-indet a with source and target $\star$, and a 2-indet r with source $a \circ a$ and target $\text{id}_\star$. All sets of higher indets are empty.



The careful reader will have noticed that there is really nothing special about mapping the single generator of $\mathbb{Z}/2\mathbb{Z}$ into corresponding free category. Indeed, this construction may be carried out for all free monoids.

Example 3.2 (Free Monoids as Computads). The free monoid functor $F_{\text{mon}} : \text{Set} \rightarrow \text{Set}$ is easily described as the functor sending each set X to the 1-computad $\mathcal{C}$ with a single 0-indet $\star$ and a set of 1-indets given by $\mathcal{C}_1 = X$.

An immediate fact about the free monoid functor F_{mon} is that it is not representable. This is because it sends the terminal set 1 to the free monoid with one generator, i.e. $\mathbb{N}$, and thus does not preserve all limits. Writing $[n]$ for the set $\{0, \dots, n\}$, we can (somewhat artificially) unravel this mapping $1 \mapsto \mathbb{N}$ as follows,

$$F_{\text{mon}}(1) \cong \mathbb{N} \cong \coprod_{i \in \mathbb{N}} 1 \cong \coprod_{i \in \mathbb{N}} \text{Set}([n], 1)$$

The inner isomorphisms are very ad-hoc, but the outer ones hold in generality. Indeed, the underlying set of a free monoid on some X is the collection of X -words of all finite lengths, and a word of length n is just a function from $[n]$ to X . Thus

$$F_{\text{mon}}(X) \cong \coprod_{n \in \mathbb{N}} \text{Set}([n], X)$$

We generalise this notion of “*very-near representability*” [CJ95, §2] to

Definition 3.3 (Familiarly Representable [CJ95, §2]). Let $\mathbb{C}$ a locally small category. We say a functor $T : \mathbb{C} \rightarrow \mathbf{Set}$ is **familiarly representable** if there exists a family $(A_i)_{i \in I}$ of objects of $\mathbb{C}$ such that T is isomorphic to a coproduct of representables

$$T(X) \cong \coprod_{i \in I} \mathbb{C}(A_i, X).$$

Clearly, the free monoid functor F_{mon} is familiarly representable by our discussion after Example 3.2. The link between the notion of familiar representability of a particular functor and the theory of computads was established by Carboni and Johnstone in [CJ95]. We will employ the machinery introduced there in Section 4.2 to recast the construction of computads under a different light, in order to answer questions around whether $\mathbf{Comp}_n$ is a presheaf category. For now, we will satisfy ourselves with introducing the relevant functor and proving its familiar representability.

Definition 3.4 (ParCell_n). For each $n \in \mathbb{N}$, we define the (**computad**) n -**parallel cell functor** $\text{ParCell}_n : \mathbf{Comp}_n \rightarrow \mathbf{Set}$ by the composition of the free n -category functor the functor constructed in Definition 2.8.

$$\mathbf{Comp}_n \xrightarrow{F_n} \mathbf{Cat}_n \xrightarrow{\text{ParCell}_n} \mathbf{Set}$$

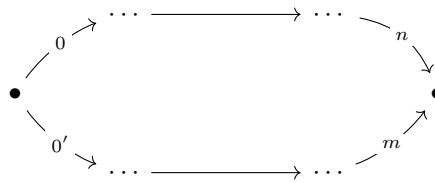
For $n = 0$, the situation is quite uninteresting:

Example 3.5. Since any 0-indets are parallel, ParCell_0 is represented by any two element set $\mathbf{2}$, and thus is trivially familiarly representable.

The next dimension, however, does require a bit of work:

Proposition 3.6 ([CJ95, 4.6]). ParCell_1 is familiarly representable.

Proof. A morphism of globular sets between the 1-sphere $\partial \mathbf{O}_1$ and a free 1-category $F_1(\mathcal{C})$ on a computad corresponds to a pair of parallel 1-cells of $F_1(\mathcal{C})$. But these are in turn given by formal composites of 1-indets of $\mathcal{C}$ of various lengths, i.e. morphisms from computads $\mathcal{D}_{n,m}$ of form



Hence

$$\frac{\frac{\mathcal{C} \mapsto \text{ParCell}_1(\mathcal{C})}{\mathcal{C} \mapsto \text{ParCell}_1(F_1(\mathcal{C}))} \text{ Definition}}{\mathcal{C} \mapsto \coprod_{n,m \in \mathbb{N}} \mathbf{Comp}_1(\mathcal{D}_{n,m}, \mathcal{C})} \text{ Above argument}$$

□

3.2 The Eckmann-Hilton Argument

As we saw in Example 3.2, encoding free monoids as free categories generated by computads is relatively trivial. On the face of it, encoding commutative monoids as free categories generated by computads should not be possible, as it requires strict equality relations between generators. This Section will be dedicated to showing that commutativity in monoids can in fact be described by leveraging the strict composition laws for ω -categories. This construction will yield our examples of non-geometric behaviour for computads, which we will explore in Section 4.4.

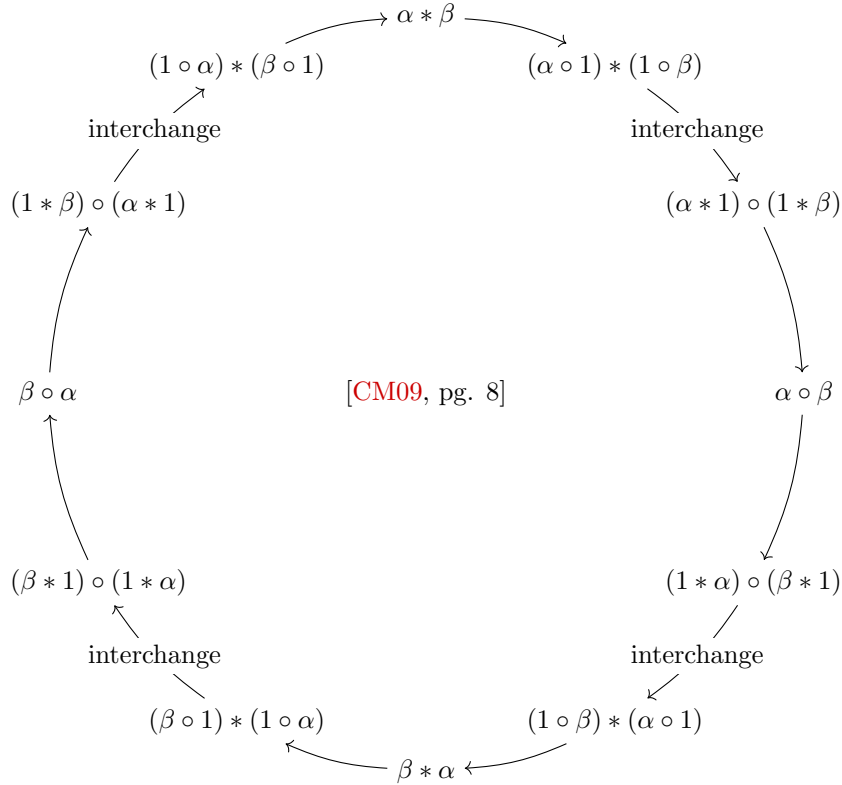
In order to prove this claim, we turn to the Eckmann-Hilton argument, which represents a characteristic higher-categorical result. This argument first appeared in [EH61, 1.12] and then again in [EH62, 4.17], from which it gained its name. It has its humble origins as a result of universal algebra, concerning sets equipped with two interacting monoid structures. In the name of elegance, we will do the reader the courtesy of presenting the result as stated in [Lei03] and as proved in [CM09], through the delightful Cheng's *clock proof*. Our presentation can be reconciled with the original account by unwinding several terse pages of definitions.

Theorem 3.7 (Eckmann-Hilton [Lei03, 1.2.4]). Let $\circ, * : M \times M \rightarrow M$ be associative operations on a set M , sharing a two-sided unit and satisfying the interchange condition

$$\forall \alpha, \beta, \alpha', \beta' \in M. (\alpha \circ \beta) * (\alpha' \circ \beta') = (\alpha * \alpha') \circ (\beta * \beta').$$

Then $\circ = *$ and $\circ$ is commutative.

Proof. Let $\alpha, \beta \in M$ and 1 be the unit of $\circ, *$. We follow the clock



□

Remarkably, a version of Theorem 3.7 holds also with weak interchange, that is with equalities being replaced by isomorphisms. However, this weaker version only gives commutativity of $\circ$ and equality with $*$ up to isomorphism, and not on the nose. This is evident from the clock argument.

Let us demonstrate a powerful application of the Eckmann-Hilton argument in homotopy theory. Recall that given a pointed topological space $(X, x_0) \in \mathbf{Top}_*$, we call $\pi_2(X, x_0)$ the quotient of the space of continuous functions $f : I^2 \rightarrow X$, by homotopies satisfying $f_t(\partial I^2) = x_0$. We may define a monoid, and in fact a group, multiplication on $\pi_2(X, x_0)$ in two ways, graphically depicted as follows:

$$\begin{array}{ccc}
 \boxed{f} & * & \boxed{g} \quad \longmapsto \quad \boxed{\begin{array}{|c|} \hline f \\ \hline g \end{array}} \\
 \\
 \boxed{f} & \circ & \boxed{g} \quad \longmapsto \quad \boxed{\begin{array}{c} f \\ \hline g \end{array}}
 \end{array}$$

Note that the boundaries are mapped to x_0 . It is evident that both operations $\circ$ and $*$ share a unit, given by the map sending I^2 to x_0 . But this means that we can apply the Eckmann-Hilton argument to $\pi_2(X, x_0)$ with respect to $\circ$ and $*$:

Corollary 3.8. Let $(X, x_0) \in \mathbf{Top}_*$ be a pointed space. Then $\pi_2(X, x_0)$ is abelian.

Proof. Following [Hat02, pg. 340], the first quadrant of the clock is given by

$$\boxed{\begin{array}{|c|} \hline f \\ \hline g \end{array}} \simeq \boxed{\begin{array}{cc} \boxed{f} & \boxed{g} \end{array}} \simeq \boxed{\begin{array}{c} \boxed{f} \\ \boxed{g} \end{array}} \simeq \boxed{\begin{array}{c} f \\ \hline g \end{array}}$$

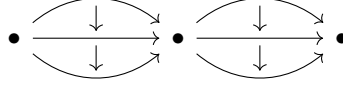
□

In category theory literature, the Eckmann-Hilton argument is usually presented as a result on monoidal categories. This is done for example in [Lei03, 1.2]. Other noteworthy forms of the argument can be found in type theoretical approaches to synthetic homotopy theory literature, such as HoTT [Uni13, 2.1.6] and CaTT [Ben18, FRV21].

3.3 Free Commutative Monoids

The Eckmann-Hilton argument is a powerful result for our study of computads: if we can find two interacting compositions operations on the cells of (free categories on) computads, we immediately get strict commutativity. An examination of Definition 2.9 reveals that the only composition of 1-cells is $\circ_0$. However, 2-cells do admit, alongside the “horizontal composition” $\circ_0$, a “vertical composition” $\circ_1$.

Furthermore, the axioms of a strict n -category given in Definition 2.9, also impose a strict interchange condition on pasting diagrams of shape



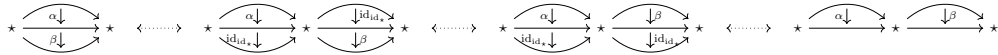
In order to obtain 2-cells that may be composed in such a fashion, we should ensure that sources and targets of both are compatible. A trivial way to get this is by having a single 0-cell and no non-identity 1-cells, which leads us to the notion of a degenerate computad.

Definition 3.9 (n -Degenerate Computad [Che12, 1.2]). We say that a computad is **n -degenerate**, for $n > 1$, if the set of 0-indets is a singleton and for all $0 < m < n$, the sets of m -indets are empty.

We will mostly concern ourselves with 2-degenerate computads, as minimal examples of the non-geometric behaviour we wish to illustrate. However, for general arguments, it is important to note that similar constructions may be carried at any higher dimensions. Turning to our 2-dimensional case, it is evident that any 2-cells of a 2-degenerate computad will necessarily have as source and target the identity 1-cell $\text{id}_\star$ on the unique 0-cell $\star$. We can now finally apply the Eckmann-Hilton argument:

Corollary 3.10. Let $\mathcal{C}$ be a 2-degenerate 2-computad. Then the vertical and horizontal compositions of any 2-cells $\alpha, \beta : \text{id}_\star \rightarrow \text{id}_\star$ of $\mathcal{C}$ are identical and commutative.

Proof. Following [Ben18, pg. 4], the first quadrant of the clock is given by



□

Let us now pause to appreciate how profoundly counter-intuitive the contents of our Corollary 3.10 are. Not only are the horizontal and vertical composition operations *strictly commutative*, they're also *strictly equal*! This behaviour is highly non-geometrical, rendering any attempt of placing an orientation on diagrams futile. Indeed, most of the confusion in computad literature arises exactly from this phenomena.

We extract an immediate consequence:

Corollary 3.11 ([Lei03, 7.6.1]). Let Compd_2 be the subcategory of 2-degenerate 2-computads. Then Compd_2 is equivalent to Set and the functor $\text{ParCell}_2 : \text{Compd}_2 \rightarrow \text{Set}$ is isomorphic to the square $F_{\text{cmon}} \times F_{\text{cmon}}$ of the free commutative monoid functor.

Proof. The equivalence of categories is induced by the restriction of the functor Cell_2 to Compd_2 . In particular, faithfulness is given by the sets of 0- and 1-indets being fixed.

Recall that any 2-cells of a 2-degenerate 2-computad $\mathcal{C}$ are parallel. Moreover, by Corollary 3.10, the operation $\circ_1$ on any 2-cells of the free 2-category on $\mathcal{C}$ is commutative. Putting everything together, we get

$$\begin{array}{lcl}
 \mathcal{C} \mapsto \text{ParCell}_2(\mathcal{C}) & & \\
 \hline
 \mathcal{C} \mapsto \text{Cell}_2(\mathcal{C}) \times \text{Cell}_2(\mathcal{C}) & \text{Degeneracy} & \\
 \hline
 \mathcal{C}_2 \mapsto \text{Cell}_2(F_2(\mathcal{C}_2)) \times \text{Cell}_2(F_2(\mathcal{C}_2)) & \text{Compd}_2 \cong \text{Set} & \\
 \hline
 \mathcal{C}_2 \mapsto F_{\text{cmon}}(\mathcal{C}_2) \times F_{\text{cmon}}(\mathcal{C}_2) & \text{Corollary 3.10} &
 \end{array}$$

□

Having established the link between commutative monoids and 2-degenerate computads, our aim will be to leverage our understanding of the functor F_{cmon} to extract some important consequences for the theory of computads. In particular, the failure of familiar representability of ParCell_2 , whose restriction to Compd_2 is related to F_{cmon} by Corollary 3.11, will play a key role in characterising the category Comp_3 in Section 4.4.

Unfortunately, at this point, we cannot yet prove that F_{cmon} is not familiarly representable. Intuitively, it should be clear that showing that no representing family can exist is a much harder task than writing out an explicit representing family. We can, however, exhibit now the behaviour that causes F_{cmon} not to be familiarly representable, and delay the proof of why that translates in a lack of familiar representability until Section 4.4.

To this end, let us conclude this Section with the following Example by Zawadowski and Makkai [MZ01], as presented by Leinster:

Example 3.12 ([Lei03, 4.1.5]). F_{cmon} does not preserve the pullback given by

$$\begin{array}{ccc} \{(x, y), (x, y'), (x', y), (x', y')\} & \xrightarrow{\pi_2} & \{y, y'\} \\ \pi_1 \downarrow & \lrcorner & \downarrow \\ \{x, x'\} & \xrightarrow{\quad} & \{z\} \end{array}$$

Indeed, for $(x, y) \cdot (x', y'), (x', y) \cdot (x, y') \in F_{\text{cmon}}\{(x, y), (x, y'), (x', y), (x', y')\}$, both projections onto $F_{\text{cmon}}\{x, x'\}$ and $F_{\text{cmon}}\{y, y'\}$ coincide

$$\begin{aligned} \pi_1((x, y) \cdot (x', y')) &= x \cdot x' = \pi_1((x', y) \cdot (x, y')), \\ \pi_2((x, y) \cdot (x', y')) &= y \cdot y' = \pi_2((x', y) \cdot (x, y')), \end{aligned}$$

but $(x, y) \cdot (x', y') \neq (x', y) \cdot (x, y')$.

Finally, we immediately translate Example 3.12 in terms of computads:

Corollary 3.13. ParCell_2 does not preserve pullbacks.

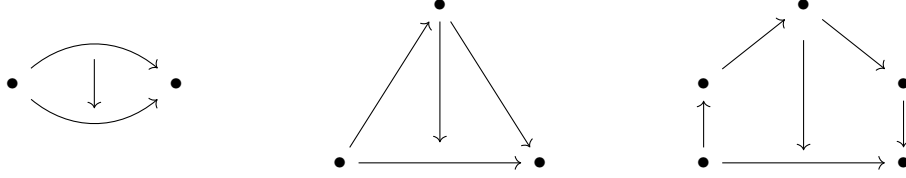
Proof. By Corollary 3.11, $\text{Compd}_2 \cong \text{Set}$, and thus is closed under pullbacks. By the same equivalence of categories, Example 3.12 may be encoded as 2-degenerate computads. Finally, since F_{cmon} does not preserve the pullback, neither will $F_{\text{cmon}} \times F_{\text{cmon}}$, and thus ParCell_2 . $\square$

4 Presheaves and Strictness

4.1 Shapes and Strictness

Having introduced strict higher categories and computads in Section 2, and the Eckmann-Hilton argument in Section 3, this final Section will focus on a comparative study of the categories $\mathbf{Comp}_2$ and $\mathbf{Comp}_3$. We will prove a deep result by Carboni and Johnstone which allow us determine the conditions under which $\mathbf{Comp}_n$ is a presheaf category. Our efforts will culminate in a proof that, when $n > 2$, $\mathbf{Comp}_n$ fails to be a presheaf category due to the Eckmann-Hilton argument. We will conclude our discussion with an example by Cheng of the failure of mere Cartesian closed-ness for $\mathbf{Comp}_3$ [Che12].

These results should be quite surprising, given that presheaves underpin most notions of higher categories, especially those defined through nerves [CL04, 1.2.3]. Indeed, no notion of “*underlying cell shape*”, be it globular, simplicial or opetopic,



can be meaningfully given to n -computads for $n > 2$. This such profoundly non-geometric behaviour is far from ideal for something we wish to take as “*the precise notion of a higher-dimensional categorical diagram*” [Mak05, (B)]!

Although the precise construction of a treacherous 3-computad will have to wait until our final Example 4.13, we already hint at the main idea behind it. Our Corollary 3.10 allowed us to find degenerate 2-computads with a commutative 2-cell composition. In Example 4.13, those non-geometric 2-cells will become the source and targets for 3-cells of our (3-)computads. This will give us indets that are “*formally distinct*” but “*algebraically undistinguishable*”, achieving an effect similar to that of Example 3.12. Let us stress again that the ultimate culprit here is strictness: an apparently “*reasonable*” assumption to place on ω -categories, which percolates through the Eckmann-Hilton clock to give us equality on the nose of cells that should rightfully only be isomorphic.

4.2 Computads by Artin Glueing

In order to realise the aforementioned pathological behaviour, we must first reinterpret our construction of computads from a new perspective. These will be the contents of Lemma 4.2, which expresses $(n + 1)$ -computads as the Artin glueing of n -computads along the functor $\mathbf{ParCell}_n$. Let us formalise this key notion:

Definition 4.1 (Artin Glueing [CJ95, §4]). Let $\mathbb{C}, \mathbb{D}$ be categories, and $F : \mathbb{C} \rightarrow \mathbb{D}$ be a functor between them. The **category obtained by (Artin) glueing along F** is the comma category $(\mathbb{D} \downarrow F)$, whose objects are the triples (A, B, f) , with $A \in \mathbb{C}, B \in \mathbb{D}$ and $f : B \rightarrow FA$ in $\mathbb{D}$, and whose morphisms $(A, B, f) \rightarrow (A', B', f')$

are the pairs $(\alpha : A \rightarrow A', \beta : B \rightarrow B')$ making the following square commute

$$\begin{array}{ccc} B & \xrightarrow{\beta} & B' \\ f \downarrow & & \downarrow f' \\ FA & \xrightarrow{F\alpha} & FA' \end{array}$$

Questions around properties of categories that arise from glueing have long attracted the interest of topos theorists. Being the study of topoi so profoundly incompatible with our concerns of brevity, we will not dwell much on how the subject came to grow from Grothendieck's work on sheaves on “*pointless topological spaces*” and Lawvere's definition of elementary topoi as categories “*axiomatically close to*” **Set**. Indeed, we will not even give a precise definition of what a topos is, instead referring the reader to the expository material in [Ill04, Rie19]. We will merely limit ourselves to hinting that topoi are, morally, “*places*” in which we can, by way of sophisticated categorical logic machinery, “*internalise*” most modern mathematics. We will also state that presheaf categories are topoi and that a key property of topoi is Cartesian closed-ness.

With this context, we can appreciate that, should the glueing of two topoi be itself a topos, this technique ought to yield a powerful way to construct non-standard mathematical universes⁴. The first result on such line of thinking is due to Artin, in the context of Grothendieck topoi [AGV72, 9.5.6]. Wraith then gave the corresponding result for elementary topoi in [Wra74], and Carboni and Johnstone gave necessary and sufficient conditions for presheaf topoi and Cartesian closed categories [CJ95]⁵. This latter paper infamously presents the category of n -computads $\mathbf{Comp}_n$ as an example of presheaf category obtained by Artin glueing. The claim that $\mathbf{Comp}_n$ can be obtained by Artin glueing is correct, and can be shown with the following argument:

Lemma 4.2 ([CJ95, 4.6]). $\mathbf{Comp}_{n+1} \cong (\mathbf{Set} \downarrow \mathbf{ParCell}_n)$.

Proof. By Definition 2.15, an object $\mathcal{C} \in \mathbf{Comp}_{n+1}$ is given by $(\mathcal{D}, (F_n(\mathcal{D}), \mathbf{A}))$ where $\mathcal{D}$ is an n -computad and $(F_n(\mathcal{D}), \mathbf{A}) \in \mathbf{Cat}_n^+$, where the elements of $\mathbf{A}(n+1)$ are the $(n+1)$ -idents of $\mathcal{C}$. This gives the natural bijection

$$\begin{array}{c} \mathcal{C} \\ \hline (\mathcal{D}, \mathbf{A}(n+1), s, t : \mathbf{A}(n+1) \rightrightarrows \mathbf{U}_n(F_n(\mathcal{D}))) \\ \hline (\mathcal{D}, \mathbf{A}(n+1), f : \mathbf{A}(n+1) \rightarrow \mathbf{ParCell}_n(\mathcal{D})) \\ \hline (\mathcal{D}, \mathcal{C}_{n+1}, f : \mathcal{C}_{n+1} \rightarrow \mathbf{ParCell}_n(\mathcal{D})) \end{array} \begin{array}{l} \text{Definition 2.15} \\ \text{ss = st, ts = tt} \\ \mathbf{A}(n+1) = \mathcal{C}_{n+1} \end{array}$$

□

Our next Section will show in detail why Carboni and Johnstone were indeed correct that $\mathbf{Comp}_2$ is a presheaf category, as first proved by Shanuel (unpublished). We will then follow it up with a thorough study of why it is not correct that this generalises to higher dimensions⁶.

⁴Artin glueing also has a topological interpretation, as a way to extend topological spaces. Although we will not touch on this, this perspective significantly shaped the original accounts of [AGV72, 9.5].

⁵For a summary of the state of the art on the subject, the reader is referred to [SA20, Table 1].

⁶Note that a corrigenda [CJ04] to [CJ95], which acknowledges this mistake has been since issued by the authors.

4.3 2-Computads are Presheaves

We will report in this Section a proof of the main result of Carboni and Johnstone’s [CJ95], which allows to determine whether a category obtained by the Artin glueing of presheaf categories is again a presheaf category. We must remark that, in our account of this result, we deviate from both our main sources [CJ95] and [Lei03], in as much as our results are as stated in [CJ95], but proven directly as sketched in [Lei03]. The reasoning behind this choice is as follows: the main result of [CJ95] relates properties of the glued category $(\widehat{\mathbb{D}} \downarrow \mathbb{T})$ to the functor $\mathbb{T} : \widehat{\mathbb{C}} \rightarrow \widehat{\mathbb{D}}$ preserving a specific class of limits, of “*connected shape*”. However, when the codomain $\widehat{\mathbb{D}}$ of $\mathbb{T}$ is **Set**, the result can be specialised to $\mathbb{T}$ being familiarly representable, which is easier to check. This is how the results are stated in [CJ95, 4.4,4.5], but the proofs given are very indirect and use some significant machinery.

A sketch of a direct proof of [CJ95, 4.5] is given in Appendix C.3 of [Lei03], but the result given is removes the need to reduce to the case $\widehat{\mathbb{D}} = \mathbf{Set}$. This is because the generalised notion of familiar representability for “*presheaf-valued functors*” $\mathbb{T} : \widehat{\mathbb{C}} \rightarrow \widehat{\mathbb{D}}$, found in [Lei03, C.3.4], is exactly equivalent to familiar representability of each $\mathbb{T}(-)(b) : \widehat{\mathbb{C}} \rightarrow \mathbf{Set}$ by taking (co)limits point-wise. This gives both an easier condition to check and an explicit construction of the sites of $(\widehat{\mathbb{D}} \downarrow \mathbb{T})$, at the cost of some burdensome notation to keep track of functorial changes in the representing family for $\mathbb{T} : \widehat{\mathbb{C}} \rightarrow \widehat{\mathbb{D}}$.

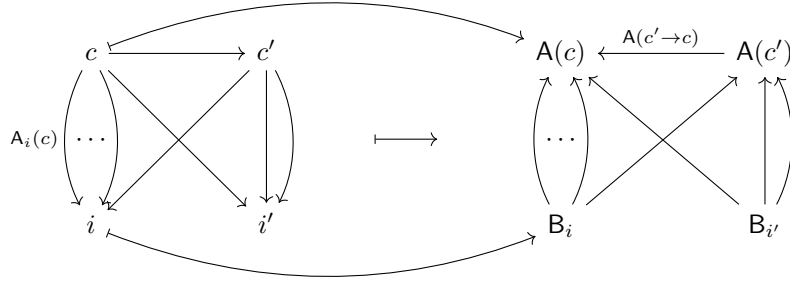
For what concerns our discussion of computads, this general version is not needed, as the case where $\mathbb{D}$ is the terminal category suffices. We will thus present an extended version of the sketch of direct proof given in [Lei03, C.3.4], restricted to our needs. Moreover, we will delay our discussion of connected limits until Section 4.4 with Theorem 4.11, where we will use that characterisation to prove $\mathbf{ParCell}_2$ is not familiarly representable.

Theorem 4.3 ([CJ95, 4.4(v)]). Let $\mathbb{T} : \widehat{\mathbb{C}} \rightarrow \mathbf{Set}$ be a functor from a presheaf category. Then $(\mathbf{Set} \downarrow \mathbb{T})$ is a presheaf category iff $\mathbb{T}$ is familiarly representable.

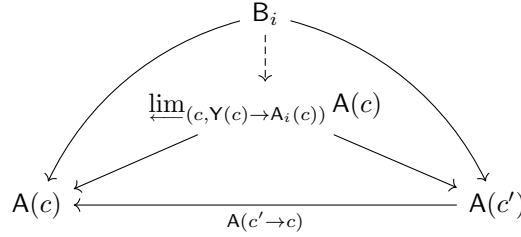
Proof. We only give the if direction, following [Lei03, C.3.4]. Assume $\mathbb{T}$ is familiarly representable and let $(A_i)_{i \in I}$ be its representing family. Consider now an arbitrary object $(A, B, f) \in (\mathbf{Set} \downarrow \mathbb{T})$. Our most important data is f , since it “*remembers the objects*”, so we will try to extract the crucial presheaf data directly out of it. We do this by chasing down the following sequence of natural isomorphisms:

$$\begin{array}{c}
 \frac{B \rightarrow \mathbb{T}(A)}{B \rightarrow \coprod_{i \in I} \widehat{\mathbb{C}}(A_i, A)} \quad \mathbb{T} \text{ Fam. Rep.} \\
 \frac{\coprod_{i \in I} B_i \rightarrow \coprod_{i \in I} \widehat{\mathbb{C}}(A_i, A)}{(B_i \rightarrow \widehat{\mathbb{C}}(A_i, A))_i} \quad B \cong \coprod_i B_i \\
 \frac{(B_i \rightarrow \widehat{\mathbb{C}}(A_i, A))_i}{(B_i \rightarrow \widehat{\mathbb{C}}(\text{colim}_{(c, Y(c) \rightarrow \mathbb{A}_i)} Y(c), A))_i} \quad f_i = f|_{B_i} \\
 \frac{(B_i \rightarrow \widehat{\mathbb{C}}(\text{colim}_{(c, Y(c) \rightarrow \mathbb{A}_i)} Y(c), A))_i}{(B_i \rightarrow \varprojlim_{(c, Y(c) \rightarrow \mathbb{A}_i)} \widehat{\mathbb{C}}(Y(c), A))_i} \quad \text{Density} \\
 \frac{(B_i \rightarrow \varprojlim_{(c, Y(c) \rightarrow \mathbb{A}_i)} \widehat{\mathbb{C}}(Y(c), A))_i}{(B_i \rightarrow \varprojlim_{(c, Y(c) \rightarrow \mathbb{A}_i)} A(c))_i} \quad \text{Hom Cont.} \\
 \frac{(B_i \rightarrow \varprojlim_{(c, Y(c) \rightarrow \mathbb{A}_i)} A(c))_i}{(B_i \rightarrow A(c))_{i \in I, c \in \mathbb{C}, a \in A_i(c)}} \quad \Upsilon \\
 \pi_{c,a} \circ -
 \end{array}$$

We claim that this data does, in fact, give a presheaf, represented by the diagram:



Indeed, we assign a set to every object of $\mathbb{C}$ and I , whereas for morphisms, we can augment those of $\mathbb{C}$, carried by A , by the $|A_i(c)|$ -many functions $B_i \rightarrow A(c)$ produced by our natural isomorphism above. Moreover, since our morphisms $B_i \rightarrow A(c)$ are obtained by projection from a limit, compositions $B_i \rightarrow A(c') \rightarrow A(c)$ are well-defined by the relevant universal properties.



Finally, we note that our decomposition only depends on $\mathbb{C}$ and on the representing family for T , which is unique up to isomorphism. This ensures our construction does build a uniform site for every object of $(\mathbf{Set} \downarrow T)$, giving our sought-after equivalence of categories. $\square$

Note that the above construction can be easily extended to the general case of $\mathbb{D}$ being arbitrary, as done in [Lei03, C.3.4]. This, however, comes at the price of needing to handle an additional (co)limit, and thus another indexing set.

We extract an immediate Corollary of Theorem 4.3:

Corollary 4.4. $\mathbf{Comp}_2$ is presheaf category.

Proof. By Lemma 4.2, $\mathbf{Comp}_2 \cong (\mathbf{Set} \downarrow \mathbf{ParCell}_1)$, and by Proposition 3.6, $\mathbf{ParCell}_1$ is familiarly representable. Since $\mathbf{Comp}_1 \cong \mathbf{Glob}_1$ is a presheaf category, the result follows immediately by Theorem 4.3. $\square$

We must now remark that, since we know explicitly the representing family for $\mathbf{Cell}_2$, Theorem 4.3 gives us a full construction of the presheaf category $\mathbf{Comp}_2$ is equivalent to. To the author best knowledge, no description of the geometric theory that this topos classifies can be found in the literature. This may be a worthwhile line for future inquiry, on account of the interest that glued models have attracted [SA20]. Moreover, the presheaf category of $\mathbf{Comp}_1$, is well understood as the topos of directed graphs. Street's [Str00], for example, gives explicit descriptions of some internal constructions for the topos $\mathbf{Glob}_n$. This leaves some hope that our understanding of $\mathbf{Comp}_1$ will simplify the logical study of $\mathbf{Comp}_2$.

4.4 3-Computads are not Presheaves

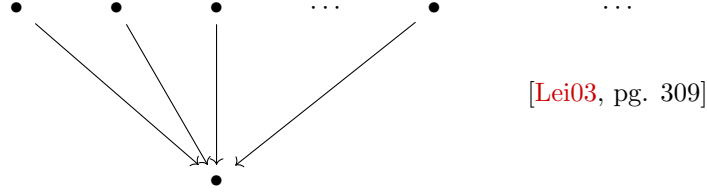
In this Section, we will finally answer in the negative the question of familiar representability of F_{cmon} , and thus ParCell_2 , which we left hanging in Section 3.3. This will allow us to show, through Theorem 4.3, that Comp_3 is not a presheaf category, counter to the claim in [CJ95]. To proceed, we will characterise familiar representability in terms of preservation of a distinguished class of limits, called “*connected limits*”. This was first done in Carboni and Johnstone’s [CJ95].

Definition 4.5 (Connected Limit [Lei03, C.2.3]). A category is **connected** if it is non-empty and any functor from it into a discrete category is constant. A **connected limit** is the limit of a diagram whose domain is a connected category.

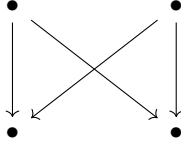
Clearly, pullbacks are an example of connected limits, and moreover, so is any “*iterated pullback*” of “*many objects*”. Such wider notion constitutes a key class of connected limits, formally defined as:

Definition 4.6 (Wide Pullback [CJ95]). A **wide pullback** is a limit indexed by a poset with a terminal object, which has all maximal chains of length 2.

Diagrammatically, the connected categories indexing wide pullbacks have shape



Example 4.7 (Non). Not every connected limit is a wide pullback. There are, for example, connected categories with no terminal object, such as



However, if our categories are presheaves, the situation is better than it appears from Example 4.7. We will now move towards proving Theorem 4.11, which shows that preservation of wide pullbacks is enough for preservation of all connected limits. Furthermore, preservation of wide pullbacks will also be shown to be equivalent to familiar representability. This will finally give us a way to prove that F_{cmon} is not familiarly representable, and extract the relevant consequences for Comp_3 .

For our arguments, it would be convenient to assume that T preserves the terminal object of $\hat{\mathbb{C}}$, which may not always be the case. Luckily, we can make use of a clever reduction argument to factor our functor T as the following composite

$$\hat{\mathbb{C}} \xrightarrow{\tilde{T}} \hat{\mathbb{D}}/T1 \xrightarrow{U} \hat{\mathbb{D}}$$

$\begin{array}{c} T \\ \curvearrowright \end{array}$

where $\tilde{T}$ is defined as

$$\begin{aligned}\tilde{T} : \widehat{\mathbb{C}} &\longrightarrow \widehat{\mathbb{D}}/\mathbb{T}1 \\ A &\longmapsto (T! : TA \rightarrow T1).\end{aligned}$$

Lemma 4.8 (Standard Reduction [CJ95, 4.2]). Let $T : \widehat{\mathbb{C}} \rightarrow \widehat{\mathbb{D}}$ be a functor between presheaf categories. Then the comma categories $(\widehat{\mathbb{D}} \downarrow T)$ and $(\widehat{\mathbb{D}}/\mathbb{T}1 \downarrow \tilde{T})$ are isomorphic.

Proof. We identify the objects $(A, B, f) \in (\widehat{\mathbb{D}} \downarrow T)$ and $(A, b, f) \in (\widehat{\mathbb{D}}/\mathbb{T}1 \downarrow \tilde{T})$ by

$$\frac{f : B \rightarrow TA}{b : B \rightarrow TA \rightarrow T1} \tilde{T}A \circ -$$

□

Thus Lemma 4.8 will allow us to swap out our T with a functor that preserves all limits, without loss of generality. This reduction will be key in proving Theorem 4.11. The final ingredients for the proof are the following two facts about connected limits, found in [Lei03], whose proof we omit:

Lemma 4.9 ([Lei03, C.2.2]). In $\mathbf{Set}$, connected limits and small products commute.

Lemma 4.10 ([Lei03, C.2.3]). For any set $I \in \mathbf{Set}$, the forgetful functor $\widehat{\mathbf{Set}}/I \rightarrow \mathbf{Set}$ preserves and reflects connected limits.

Theorem 4.11 ([Lei03, C.3.2]). Let $T : \widehat{\mathbb{C}} \rightarrow \widehat{\mathbb{D}}$ be a functor between presheaf categories. Then the following are equivalent:

- (i) T preserves connected limits,
- (ii) T preserves wide pullbacks,
- (iii) For each $b \in \mathbb{D}$, $T_b : \widehat{\mathbb{C}} \rightarrow \mathbf{Set}$ is familiarly representable,

Proof. Since limits in presheaf categories are taken point-wise, we pass without loss of generality to the case where $\mathbb{D}$ is the terminal category.

The first implication is immediate, as wide pullbacks are connected limits. Assume now that T preserves wide pullbacks. By the Standard Reduction Lemma 4.8, we can factor T as $U \circ \tilde{T}$. By Lemma 4.10, forgetful functor $U : \widehat{\mathbf{Set}}/\mathbb{T}1 \rightarrow \mathbf{Set}$ reflects wide pullbacks and T preserves them by assumption. Moreover, $\tilde{T}$ preserves the terminal object, so it preserves all limits.

Since $\widehat{\mathbb{C}}$ satisfies the Solution Set Condition of Freyd's Adjoint Functor Theorem, we have a left adjoint S to $\tilde{T}$. We get a further pair of adjoint functors $i_! \dashv (-)_i$ for every $i \in \mathbb{T}1$ acting as

$$\begin{aligned}i_! : X &\longmapsto (x \mapsto i), \\ (-)_i : (f : X \rightarrow T1) &\longmapsto f^{-1}\{i\}.\end{aligned}$$

which we can compose as

$$\widehat{\mathbb{C}} \begin{array}{c} \xrightarrow{\tilde{T}} \\ \xleftarrow[S]{T} \end{array} \widehat{\mathbf{Set}}/\mathbb{T}1 \begin{array}{c} \xrightarrow{(-)_i} \\ \xleftarrow[i_!]{T} \end{array} \mathbf{Set}$$

The representing family for $\mathbf{T}$ is now given by the natural isomorphism

$$\begin{array}{c}
\frac{A \mapsto \mathbf{T}A}{\mathbf{T} = \mathbf{U}\tilde{\mathbf{T}}} \\
\frac{A \mapsto \mathbf{U}\tilde{\mathbf{T}}A}{\mathbf{T}1 \cong \coprod_{i \in \mathbf{T}1} \{i\}} \\
\frac{A \mapsto \coprod_{i \in \mathbf{T}1} (\tilde{\mathbf{T}}A)_i}{\mathbf{Y}} \\
\frac{A \mapsto \coprod_{i \in \mathbf{T}1} \mathbf{Set}(1, (\tilde{\mathbf{T}}A)_i)}{i_! \dashv (-)_i} \\
\frac{A \mapsto \coprod_{i \in \mathbf{T}1} \mathbf{Set}/\mathbf{T}1(i_!1, \tilde{\mathbf{T}}A)}{S \dashv \tilde{\mathbf{T}}} \\
A \mapsto \coprod_{i \in \mathbf{T}1} \hat{\mathbf{C}}(Si_!1, A)
\end{array}$$

Finally, assume that $\mathbf{T}$ is familiarly representable. Then $\mathbf{T}$ preserves connected limits since representables preserve all limits and coproducts commute with connected limits by Lemma 4.9. $\square$

We must remark now that Theorem 4.11 is of limited usefulness for actually extracting the representability family out of a given wide pullback-preserving functor. This is due to the close dependency on the left adjoint $\mathbf{S}$ of $\mathbf{T}$, which is constructed by AFT. Working out explicitly how $\mathbf{S}$ acts, without any prior knowledge of the representing family for $\mathbf{T}$ is a very hard task.

We now tie everything together:

Corollary 4.12. $\mathbf{Comp}_3$ is not presheaf category.

Proof. By our Artin glueing picture established in Lemma 4.2, $\mathbf{Comp}_3 \cong (\mathbf{Set} \downarrow \mathbf{ParCell}_2)$. But by Corollary 3.13, $\mathbf{ParCell}_2$ does not preserve pullbacks. Therefore, it does not preserve wide pullbacks, and thus by Theorem 4.11, it is not familiarly representable. Finally, by Theorem 4.3, $\mathbf{Comp}_3$ is not a presheaf category. $\square$

This counter-example was first pointed out by Zawadowski and Makkai [MZ01] and acknowledged in [CJ04] as a counter-example to the aforementioned claim in [CJ95] that n -computads are a presheaf category for any n . The source of confusion for [CJ95] was an implicit choice of ordering of the indets of n -computads, which can only be done when the Eckmann-Hilton argument does not apply [CJ04]. Unfortunately, the mistake was repeated by Batanin for his generalised computads in [Bat98a, 4.1] and [Bat02b, App. C]. This was corrected in [Bat02a, 5.3] in light of [MZ01].

4.5 3-Computads are not Cartesian closed

However, the situation is actually worse than $\mathbf{Comp}_3$ just failing to be a presheaf. Recall that the pullback given in Example 3.12, from which we ultimately derived our counter-example, involves the terminal object of $\mathbf{Set}$, or equivalently $\mathbf{Comp}_2$. Hence $\mathbf{ParCell}_2$ does not even preserve binary products. By another result of Carboni and Johnstone ([CJ95, 4.1(i)]), this implies that $\mathbf{Comp}_3$ is not even Cartesian closed.

We will dedicate this final Section to showing how Cartesian closed-ness fails in $\mathbf{Comp}_3$. Although it would be theoretically possible to do so by following the proof of [CJ95, 4.1(i)], this approach would not be particularly illuminating, due to the indirect nature of the argument given in [CJ95]. Instead, we will report the direct proof given by Cheng [Che12], by way of an explicit example of a co-equaliser not being preserved under a “product by $-$ ” functor. This direct example will use much of our discussion of the Eckmann-Hilton argument in Section 3.2. All the 3-computads involved will be 2-degenerate and the key step of the following construction, namely the setup of $\mathcal{A} \times \mathcal{B}$, will be morally close to both Example 3.12 and Example 2.2.

Example 4.13 ([Che12, §2]). Consider the following 2-degenerate 3-computads:

- (i) $\mathcal{A}$ with 2-indets a_1, a_2 and 3-indet $a_1 \circ_1 a_2 \xrightarrow{f} \text{id}_{\text{id}_*}$,
- (ii) $\mathcal{B}$ with 2-indets b_1, b_2 and 3-indet $b_1 \circ_1 b_2 \xrightarrow{g} \text{id}_{\text{id}_*}$,
- (iii) $\mathcal{E}$ with 2-indets e_1, e_2 and no 3-indets,
- (iv) $\mathcal{C}$ with 2-indet c and 3-indet $c \circ_1 c \xrightarrow{h} \text{id}_{\text{id}_*}$.

Consider now the morphisms of computads $\alpha_1, \alpha_2 : \mathcal{E} \rightrightarrows \mathcal{A}$ and $\beta : \mathcal{A} \rightarrow \mathcal{C}$ given by

$$\begin{aligned} \alpha_1 : e_1 &\mapsto a_1, \quad e_2 \mapsto \text{id}_{\text{id}_*}, \\ \alpha_2 : e_1 &\mapsto a_2, \quad e_2 \mapsto \text{id}_{\text{id}_*}, \\ \beta : a_1 &\mapsto c, \quad a_2 \mapsto c, \quad f \mapsto h. \end{aligned}$$

Then the functor $- \times \mathcal{B}$ does not preserve the co-equaliser

$$\mathcal{E} \begin{array}{c} \xrightarrow{\alpha_1} \\ \xrightarrow{\alpha_2} \end{array} \mathcal{A} \xrightarrow{\beta} \mathcal{C}$$

Proof. Checking that $\mathcal{C}$ is indeed a co-equaliser amounts to a rather straight-forward check of the relevant universal property, which we omit.

$$\begin{array}{ccc} \mathcal{E} & \begin{array}{c} \xrightarrow{\alpha_1} \\ \xrightarrow{\alpha_2} \end{array} & \mathcal{A} \xrightarrow{\beta} \mathcal{C} \\ & & \searrow \gamma \quad \downarrow \exists! \\ & & \mathcal{D} \end{array}$$

The functor $- \times \mathcal{B}$ acts on the co-equaliser diagram above to give

$$\mathcal{E} \times \mathcal{B} \begin{array}{c} \xrightarrow{\alpha_1 \times \text{id}} \\ \xrightarrow{\alpha_2 \times \text{id}} \end{array} \mathcal{A} \times \mathcal{B} \xrightarrow{\beta \times \text{id}} \mathcal{C} \times \mathcal{B}$$

We claim that $\mathcal{C} \times \mathcal{B}$ is not a co-equaliser. First, we wish to understand the structure of $\mathcal{A} \times \mathcal{B}$ explicitly. We claim it has 2-indets $(a_1, b_1), (a_1, b_2), (a_2, b_1), (a_2, b_2)$ and 3-indets

$$\begin{aligned} (a_1, b_1) \circ_1 (a_2, b_2) &\xrightarrow{(f,g)_1} (\text{id}_{\text{id}_*}, \text{id}_{\text{id}_*}), \\ (a_2, b_1) \circ_1 (a_1, b_2) &\xrightarrow{(f,g)_2} (\text{id}_{\text{id}_*}, \text{id}_{\text{id}_*}). \end{aligned}$$

Up to 2-cells, this is evidently true. Let $\mathcal{D}$ be any other computad admitting morphisms to $\mathcal{A}$ and $\mathcal{B}$ as in the diagram below

$$\begin{array}{ccc} & \mathcal{D} & \\ & \downarrow \exists! & \\ \varphi \swarrow & \mathcal{A} \times \mathcal{B} & \searrow \psi \\ \pi_1 \swarrow & & \searrow \pi_2 \\ \mathcal{A} & & \mathcal{B} \end{array}$$

Then any 3-indet k of $\mathcal{D}$ must be sent to f by φ and to g by ψ . Moreover, since morphisms of computads commute with source and target maps, k must be of the form

$d_1 \circ_1 d_2 \xrightarrow{k} (\text{id}_{\text{id}_*}, \text{id}_{\text{id}_*})$ for some 2-indets d_1, d_2 of $\mathcal{D}$. The Eckmann-Hilton argument causes some ambiguities with regards to where $d_1 \circ_1 d_2$ gets sent. By enumerating the options, we find that either φ sends (d_1, d_2) to (a_1, a_2) or it sends it to (a_2, a_1) , and similarly for ψ . However, in each of the four cases, we obtain one of $(f, g)_1$ or $(f, g)_2$. This proves that the cell structure given for $\mathcal{A} \times \mathcal{B}$ is correct. As for the other products, $\mathcal{E}$ has no 3-indets, so neither does $\mathcal{E} \times \mathcal{B}$. For $\mathcal{C} \times \mathcal{B}$, we may similar check that the 2-indets are given by $(c, b_1), (c, b_2)$ and the only 3-indet by $(c, b_1) \circ_1 (c, b_2) \xrightarrow{(h, g)} \text{id}_{\text{id}_*}$.

We now claim that the co-equaliser of $\mathcal{E} \times \mathcal{B} \rightrightarrows \mathcal{A} \times \mathcal{B}$ is not $\mathcal{C} \times \mathcal{B}$. To this end, consider the computad $\mathcal{P}$ with 2-indets $(c, b_1), (c, b_2)$ and two 3-indets

$$\begin{aligned} (c, b_1) \circ_1 (c, b_2) &\xrightarrow{(f, g)_1} (\text{id}_{\text{id}_*}, \text{id}_{\text{id}_*}), \\ (c, b_1) \circ_1 (c, b_2) &\xrightarrow{(f, g)_2} (\text{id}_{\text{id}_*}, \text{id}_{\text{id}_*}), \end{aligned}$$

and the morphism $\mathcal{A} \times \mathcal{B} \xrightarrow{\gamma} \mathcal{P}$, which acts on 2-indets by sending (a_i, b_j) to (c, b_j) and on 3-indets by sending $(f, g)_i$ to $(f, g)_i$.

$$\begin{array}{ccccc} \mathcal{E} \times \mathcal{B} & \xrightleftharpoons[\alpha_2]{\alpha_1} & \mathcal{A} \times \mathcal{B} & \xrightarrow{\beta} & \mathcal{C} \times \mathcal{B} \\ & & \searrow \gamma & & \downarrow \exists \\ & & & & \mathcal{P} \end{array}$$

Clearly, $\gamma\alpha_1 = \gamma\alpha_2$, but there is no way to factor γ through $\mathcal{C} \times \mathcal{B}$, as it only has a single 3-indet. Since $\mathcal{P}$ is evidently not isomorphic to $\mathcal{C} \times \mathcal{B}$, the co-equaliser is not preserved. Hence $- \times \mathcal{B}$ has no right adjoint. $\square$

Note that the above Example 4.13 generalises quite trivially to higher dimensions, and even to ω -computads, by suitably lifting the degenerate computads involved.

Let us extract a final consequence:

Corollary 4.14 ([MZ08, 3.2]). Comp_ω and Comp_n for $n > 2$ are not locally Cartesian closed.

Proof. Let $n > 2$. By Example 4.13, Comp_ω and Comp_n are not Cartesian closed. Since both categories have a terminal object, we have

$$\begin{aligned} \text{Comp}_\omega/1 &\cong \text{Comp}_\omega, \\ \text{Comp}_n/1 &\cong \text{Comp}_n. \end{aligned}$$

Thus at least one of their slices fails to be Cartesian closed. $\square$

Example 4.13 is a fitting culmination for our study of computads, even just by virtue of the sheer number of cells involved. To prevent the key idea from being drowned by noise, we will point out for the reader that the pivotal point of the argument lies in the discovery that $\mathcal{A} \times \mathcal{B}$ has *two* 3-indets. This is not at all intuitive before one checks the universal property, and has the direct consequence of making $\mathcal{C} \times \mathcal{B}$ fail to be a co-equaliser. Thus, our direct Example 4.13 trades the need for the heavy theoretical machinery of Theorems 4.3 and 4.11, for several tedious checks of universal properties.

We leave to the reader's taste the final choice as to which argument flows better.

5 Conclusion

The theory of strict computads is unique in its ability to exemplify many key characteristic features of higher category theory. Computads show how the quest for a formal theory of higher dimensional categorical diagrams can help us navigate the complex tangle of abstractions that surrounds higher categories. The ease with which familiar mathematical structures may be viewed as computads highlights genuine need for a rigorous theory of higher categories.

At the same time, computads show how the assumption of strictness, innocuous in one dimension, can lead to unexpected pathological behaviour when passing to higher dimensions. The degeneracies introduced by strictness are so severe that they end up hampering our ability reason geometrically about diagrams.

In order not to leave the reader excessively dismayed after our barrage of counterexamples, we will point them towards two lines of active research on how to best remedy the failures discussed. As we have come to appreciate in the conclusive part of our discussion, computads form a presheaf category if and only if the functor $\mathbf{ParCell}$ is familiarly representable. One way to ensure this is the case is to rule out degenerate computads, for which Eckmann-Hilton argument may be applied, by passing to a suitable subcategory of $\mathbf{Comp}$. The fact that no other types of degeneracies are preventing computads from being presheaves was first conjectured in the closing remarks of [CJ04]. Recent work by Henry has offered a proof that this conjecture does indeed hold true, alongside a significant theory for detecting which classes of computads are free of pathological behaviour [Hen18].

The other way to remedy the failure of familiar representability of $\mathbf{ParCell}$ is to give up on the requirement for strictness and consider weak categories, where unit cancellation, interchange or associativity do not hold on the nose, but only up to suitably-invertible higher cells. Significant work on the subject has been done by Batanin [Bat98a, Bat02a], who introduced a notion of computads parametrised by monads. His definition coincides with the one reported here when the algebras for the given monad are strict n - or ω -categories, but it does also allow for computads on weak higher categories, by suitable choice of monads. The extreme flexibility of Batanin's parametric computads does, however, make them considerably more bureaucratic than their strict counterparts. A solution to this problem may very well come from type theoretic approaches to weak higher categories [FM17, FRV21], which would generalise the syntactic study of computads of [Pen99, Mak05] beyond the strict case⁷.

I would like to thank Dr. Jamie Vicary and Ioannis Markakis for their invaluable advice on the topic.

⁷The author is aware of unpublished research on the topic, but is unable to point to any extensive published account on the matter, beyond what already discussed.

References

- [AGV72] M. Artin, A. Grothendieck, and J. L. Verdier, *Topos*, Théorie des Topos et Cohomologie Etale des Schémas, 1972, pp. 299–518 (fr).
- [Bat02a] M. A. Batanin, *Computads and slices of operads*, arXiv:math (September 2002). arXiv: math/0209035.
- [Bat02b] ———, *On the Penon method of weakening algebraic structures*, Journal of Pure and Applied Algebra **172** (July 2002), no. 1, 1–23 (en).
- [Bat98a] ———, *Computads for finitary monads on globular sets*, Contemporary Mathematics, 1998, pp. 37–57 (en).
- [Bat98b] M.A. Batanin, *Monoidal Globular Categories As a Natural Environment for the Theory of Weakn-Categories*, Advances in Mathematics **136** (June 1998), no. 1, 39–103 (en).
- [Ben18] Thibaut Benjamin, *Towards a fully formalized definition of syllepsis in weak higher categories*, Proceedings of HDRA workshop, 2018, pp. 10.
- [Bur91] Albert Burroni, *Higher dimensional word problem*, Category Theory and Computer Science, 1991, pp. 94–105 (en). Series Title: Lecture Notes in Computer Science.
- [Bur93] ———, *Higher-dimensional word problems with applications to equational logic*, Theoretical Computer Science **115** (July 1993), no. 1, 43–62 (en).
- [CG07] Eugenia Cheng and Nick Gurski, *The periodic table of n -categories for low dimensions I: degenerate categories and degenerate bicategories*, arXiv:0708.1178 [math] (August 2007) (en). arXiv: 0708.1178.
- [Che12] Eugenia Cheng, *A direct proof that the category of 3-computads is not cartesian closed*, arXiv:1209.0414 [math] (December 2012). arXiv: 1209.0414.
- [CJ04] Aurelio Carboni and Peter Johnstone, *Corrigenda for Connected limits, familial representability and Artin glueing*, Mathematical Structures in Computer Science **14** (February 2004), no. 1, 185–187 (en).
- [CJ95] ———, *Connected limits, familial representability and Artin glueing*, Mathematical Structures in Computer Science **5** (December 1995), no. 4, 441–459 (en).
- [CL04] Eugenia Cheng and Aaron Lauda, *Higher-Dimensional Categories: An Illustrated Guide Book* (2004), 182 (en).
- [CM09] Eugenia Cheng and Michael Makkai, *A note on the Penon definition of n -category*, arXiv:0907.3961 [math] (July 2009) (en). arXiv: 0907.3961.
- [EH61] B. Eckmann and Peter Hilton, *Structure maps in group theory*, Fundamenta Mathematicae **50** (1961), 207–221 (en). Publisher: Instytut Matematyczny Polskiej Akademii Nauk.
- [EH62] B. Eckmann and P. J. Hilton, *Group-like structures in general categories I multiplications and comultiplications*, Mathematische Annalen **145** (June 1962), no. 3, 227–255 (en).
- [FM17] Eric Finster and Samuel Mimram, *A Type-Theoretical Definition of Weak ω -Categories*, arXiv:1706.02866 [cs, math] (June 2017) (en). arXiv: 1706.02866.
- [FRV21] Eric Finster, Alex Rice, and Jamie Vicary, *A Type Theory for Strictly Associative Infinity Categories*, arXiv:2109.01513 [cs, math] (September 2021). arXiv: 2109.01513.
- [GM18] Yves Guiraud and Philippe Malbos, *Polygraphs of finite derivation type*, Mathematical Structures in Computer Science **28** (February 2018), no. 2, 155–201. arXiv: 1402.2587.
- [Gue21] Léonard Guetta, *Homology of categories via polygraphic resolutions*, Journal of Pure and Applied Algebra **225** (October 2021), no. 10, 106688 (en).
- [Hat02] Allen Hatcher, *Algebraic topology*, University Press, Cambridge, 2002 (eng). Book Title: Algebraic topology / Allen Hatcher.
- [Hen18] Simon Henry, *Non-unital polygraphs form a presheaf category*, arXiv:1711.00744 [math] (September 2018) (en). arXiv: 1711.00744.
- [Hil90] David Hilbert, *Ueber die Theorie der algebraischen Formen*, Mathematische Annalen **36** (December 1890), no. 4, 473–534 (de).
- [Ill04] Luc Illusie, *What Is... a Topos?*, Notices of the American Mathematical Society **51** (October 2004), no. 9, 1060–1061.
- [Joh89] Michael Johnson, *The combinatorics of n -categorical pasting*, Journal of Pure and Applied Algebra **62** (1989), 211–225.
- [KB83] D. E. Knuth and P. B. Bendix, *Simple Word Problems in Universal Algebras*, Automation of Reasoning, 1983, pp. 342–376 (en).

- [Lei00] Tom Leinster, *Operads in Higher-Dimensional Category Theory*, arXiv:math (November 2000). arXiv: math/0011106.
- [Lei01] ———, *A Survey of Definitions of n -Category*, arXiv:math (July 2001) (en). arXiv: math/0107188.
- [Lei03] ———, *Higher Operads, Higher Categories*, arXiv:math (May 2003). arXiv: math/0305049.
- [Lei98] ———, *Structures in higher-dimensional category theory*, arXiv:math (October 1998). arXiv: math/0109021.
- [LLR89] René Lavendhomme, Thierry Lucas, and Gonzalo Reyes, *Formal systems for topos-theoretic modalities*, Bulletin de la Société Mathématique de Belgique, Série A **41** (1989), 333.
- [Mé03] François Métayer, *Resolutions by Polygraphs*, Theory and Applications of Categories **11** (May 2003), no. 7, 148–184. Publisher: Mount Allison University.
- [Mé07] ———, *Cofibrant complexes are free*, arXiv:math (January 2007) (en). arXiv: math/0701746.
- [Mé16] ———, *Strict ω -categories are monadic over polygraphs*, arXiv:1606.00139 [math] (June 2016) (en). arXiv: 1606.00139.
- [Mak05] Michael Makkai, *The word problem for computads* (2005).
- [Mak07] ———, *Computads and 2 dimensional pasting diagrams* (April 2007). Publisher: Citeseer.
- [MZ01] Michael Makkai and Marek Zawadowski, *3-computads do not form a presheaf category* (2001), 2 (en).
- [MZ08] ———, *The category of 3-computads is not cartesian closed*, Journal of Pure and Applied Algebra **212** (November 2008), no. 11, 2543–2546 (en).
- [Pen99] Jacques Penon, *Approche polygraphique des ∞ -catégories non strictes*, Cahiers de Topologie et Géométrie Différentielle Catégoriques **40** (1999), no. 1, 31–80 (fra). Publisher: Dunod éditeur, publié avec le concours du CNRS.
- [Rey91] Gonzalo E. Reyes, *A topos-theoretic approach to reference and modality.*, Notre Dame Journal of Formal Logic **32** (June 1991), no. 3, 359–391. Publisher: Duke University Press.
- [Rie19] Emily Riehl, *Sketches of an Elephant: An Introduction to Topos Theory* (September 2019), 8 (en).
- [RR04] Marie La Palme Reyes and Gonzalo E Reyes, *Generic figures and their glueings* (June 2004), 267 (en).
- [RZ93] Gonzalo E. Reyes and Marek W. Zawadowski, *Formal systems for modal operators on locales*, Studia Logica **52** (December 1993), no. 4, 595–613 (en).
- [SA20] Jonathan Sterling and Carlo Angiuli, *Gluing Models of Type Theory along Flat Functors* (January 2020), 35 (en).
- [SII12] D Singh, A I Isah, and A M Ibrahim, *A Survey of the Development of the Theory of Categories*, International Journal of Innovation **3** (2012), no. 2, 4 (en).
- [SOK94] Craig C. Squier, Friedrich Otto, and Yuji Kobayashi, *A finiteness condition for rewriting systems*, Theoretical Computer Science **131** (September 1994), no. 2, 271–294 (en).
- [Squ87] Craig C. Squier, *Word problems and a homological finiteness condition for monoids*, Journal of Pure and Applied Algebra **49** (November 1987), no. 1, 201–217 (en).
- [Str00] Ross Street, *The petit topos of globular sets*, Journal of Pure and Applied Algebra **154** (December 2000), no. 1-3, 299–315 (en).
- [Str76] ———, *Limits indexed by category-valued 2-functors*, Journal of Pure and Applied Algebra **8** (June 1976), no. 2, 149–181 (en).
- [Str87] ———, *The algebra of oriented simplexes*, Journal of Pure and Applied Algebra **49** (December 1987), no. 3, 283–335 (en).
- [Thi03] Rüdiger Thiele, *Hilbert’s twenty-fourth problem*, The American mathematical monthly **110** (2003), no. 1, 1–24. Publisher: Taylor & Francis.
- [Uni13] The Univalent Foundations Program, *Homotopy type theory: Univalent foundations of mathematics*, <https://homotopytypetheory.org/book>, Institute for Advanced Study, 2013.
- [Wal89] R. F. C. Walters, *Data types in distributive categories*, Bulletin of the Australian Mathematical Society **40** (August 1989), no. 1, 79–82 (en). Publisher: Cambridge University Press.
- [Wra74] Gavin Wraith, *Artin glueing*, Journal of Pure and Applied Algebra **4** (June 1974), no. 3, 345–348 (en).