A Type Theory for Strictly Associative Infinity Categories

Eric Finster
University of Birmingham
e.l.finster@bham.ac.uk

Alex Rice
University of Cambridge
alex.rice@cl.cam.ac.uk

Jamie Vicary
University of Cambridge
jamie.vicary@cl.cam.ac.uk

March 8, 2024

Abstract

Many definitions of weak and strict ∞-categories have been proposed. In this paper we present a definition for ∞-categories with strict associators, but which is otherwise fully weak. Our approach is based on the existing type theory Catt, whose models are known to correspond to weak ∞-categories. We add a definitional equality relation to this theory which identifies terms with the same associativity structure, yielding a new type theory Catt$_{\mathsf{sa}}$, for strictly associative ∞-categories. We also provide a reduction relation which generates definitional equality, and show it is confluent and terminating, giving an algorithm for deciding equality of terms, and making typechecking decidable.

Our key contribution, on which our reduction is based, is an operation on terms which we call *insertion*. This has a direct geometrical interpretation, allowing a subterm to be inserted into the head of the term, flatting its syntactic structure. We describe this operation combinatorially in terms of pasting diagrams, and also show can be characterized as a pushout of contexts. This allows reasoning about insertion using just its universal property.

1 Introduction

It has long been known that not every fully weak ∞-category is equivalent to one which is fully strict[14]. While every bicategory (and by extension monoidal category) is equivalent to a strict 2-category (respectively strict monoidal category), one can show that such an equivalence is not even possible at dimension 3, due to an idea involving the Eckmann-Hilton argument[9].

There are three main classes of coherence in a (globular) ∞-category: Unitors have the form

$$f \cdot \mathrm{id} \Rightarrow f$$

associators have the form

$$(f \cdot g) \cdot h \Rightarrow f \cdot (g \cdot h)$$

and interchangers have the form

$$(a \cdot_n b) \cdot_m (c \cdot_n d) \Rightarrow (a \cdot_m c) \cdot_n (b \cdot_m d)$$

for compositions $\cdot_n$ and $\cdot_m$ of different codimensions (i.e. one could be a vertical composition and the other a horizontal composition). The Eckmann-Hilton argument implies that commutativity

1

of scalars can be built from these 3 classes of coherences, and strictness of these coherences gives a strict commutativity of scalars, which in turn implies that not every weak ∞-category is equivalent to a strict one.

It is therefore clear that any definition of ∞-category in which all three of these classes is strict is not general enough to model all weak ∞-categories. However this does not mean that no structure of the category can be strictified, and this leads to the search of a semistrict definition of an ∞-category, one which is "as strict as possible" while still retaining the full expressivity of fully weak ∞-categories.

In dimension 3, two semistrict definitions have been given:

- Gray categories[6] are a definition where everything is strict apart from the interchange laws, which hold only up to isomorphism. It is proven that every tricategory is equivalent to a Gray category.

- Joyal and Kock[8] introduce a definition for monoidal 2-categories (i.e. 3-categories with one object) which have weak units, but all other structure being strict, and show this is equivalent to the fully weak version.

These suggest two approaches to defining semistrict ∞-categories, one where unitors and associators are strict and interchangers are weak, as in Gray categories, and one where everything but units are strict, which is known as Simpson's conjecture.

Although many definitions of weak ∞-categories have been given (for example see [10]), not many seem to be partially strict. Two of the authors, along with Reutter, have given a definition of a strictly unital ∞-category[5] by putting a reduction relation on the type theory Catt[4]. Our work, running in parallel to this, uses the same technique but uses a new reduction relation which quotients out by associators instead of unitors.

Catt is a type theory for which we can define the models to be ∞-categories. Recently it has been shown by Benjamin, Finster, and Mimram[2], that the models of this type theory are exactly Grothendieck-Maltsiniotis ∞-categories[12]. In contrast to the more abstract definitions, Catt gives us a way to reason about the theory inductively and naturally allows computer assistance for defining and checking terms. By giving a complete reduction (one which is strongly normalising and confluent) we can create a quotient of this theory while retaining its computational properties such as decidable equality of terms and decidable typechecking.

To define our new reduction relation, we introduce an operation which we call "insertion". This operation effectively combines compound composites into a single composite. The simplest example of this is the standard 1-dimensional associativity equation. If we start with the compound composite $f \cdot (g \cdot h)$, insertion will combine these into a single ternary composition $f \cdot g \cdot h$. Similarly the term $(f \cdot g) \cdot h$ will be reduced to the same ternary composition. We call this operation insertion as when we view this as an operation on pasting diagrams, the pasting diagram for the inner composition is inserted into the pasting diagram for the outer composition. We give relatively weak conditions for this insertion to be admissible, and further show that it satisfies a certain universal property.

1.1 Contributions and Future Work

We introduce a type theory $\mathsf{Catt_{sa}}$ for which the models form strictly associative ∞-categories. This is done by defining an operation on the structure of the type theory Catt called insertion and creating an equality relation based on this. We further prove some properties of this insertion operation,

2

and show that there is a reduction scheme which agrees with the equality and is complete, and hence give algorithms for deciding equality and typechecking.

Further we show that the pruning operation introduced for strict unit normalisation[5] can be seen as an instance of our insertion operation, which suggests the two reduction schemes are compatible and can be combined to give a type theory which models strictly associative and unital ∞-categories. Proving that this reduction scheme is indeed complete is ongoing work.

Finally, we would eventually like to be able to find a connection between our work and associative n-categories[3], which are a far more combinatoric definition of strictly associative and unital higher categories. These have the advantage that they can be viewed and manipulated graphically by contractions[13], and have been implemented in the proof assistant "homotopy.io"[7]. We eventually hope to find a strict enough type theoretic definition of higher categories such that associative n-category terms can be converted into the terms of this theory. This would have the two purposes: firstly it gives evidence that the structures manipulated by a tool such as "homotopy.io" really do correspond to conventional definitions of higher categories and secondly could make it easier to construct terms of Catt and related theories.

2 Catt

We first define the syntax and typing rules of the base type theory Catt, which models weak ∞-categories. Here we give a full account of all the rules of the type theory, and carefully set up how we will add definitional equality into the theory, though do not give much motivation for many of the base roles, for which we encourage the reader to look at the original paper on Catt[4] or Benjamin's PhD thesis[1].

2.1 Syntax

Let V be an infinite set of variables. Then the syntax of *contexts*, *substitutions*, *types*, and *terms* (written Ctx, Sub, Type, Term) is given by:

$$\frac{}{\varnothing : \mathsf{Ctx}} \qquad \frac{\Gamma : \mathsf{Ctx} \quad A : \mathsf{Type} \quad x \in V}{\Gamma, (x : A) : \mathsf{Ctx}}$$

$$\frac{}{\langle\rangle : \mathsf{Sub}} \qquad \frac{\sigma : \mathsf{Sub} \quad t : \mathsf{Term} \quad x : V}{\langle\sigma, x \mapsto t\rangle : \mathsf{Sub}}$$

$$\frac{}{\star : \mathsf{Type}} \qquad \frac{A : \mathsf{Type} \quad s : \mathsf{Term} \quad t : \mathsf{Term}}{s \to_A t : \mathsf{Type}}$$

$$\frac{x \in V}{x : \mathsf{Term}} \qquad \frac{\Gamma : \mathsf{Ctx} \quad A : \mathsf{Type} \quad \sigma : \mathsf{Sub}}{\mathsf{coh}\ (\Gamma : A)[\sigma] : \mathsf{Term}}$$

The symbol $\equiv$ will represent syntactic equality.

We define the *support* of a term, type, or substitution in context Δ as follows:

- Substitutions: $\mathsf{supp}_\Delta(\langle\rangle) = \varnothing$, $\mathsf{supp}_\Delta(\langle\sigma, t\rangle) = \mathsf{supp}_\Delta(\sigma) \cup \mathsf{supp}_\Delta(t)$.

- Types: $\mathsf{supp}_\Delta(\star) = \varnothing$, $\mathsf{supp}(s \to_A t) = \mathsf{supp}_\Delta(s) \cup \mathsf{supp}_\Delta(t)$

3

- Terms:

$$\begin{aligned}
\mathsf{supp}_\Delta(x) &= \{x\} \cup \mathsf{supp}_\Delta(A) \quad \text{if } (x:A) \in \Delta \\
\mathsf{supp}_\Delta(\mathsf{coh}\ (\Gamma:A)[\sigma]) &= \mathsf{supp}_\Delta(\sigma)
\end{aligned}$$

We Write $\mathsf{supp}(t)$ when the context is obvious.

A dimension function dim on types is defined by:

$$\begin{aligned}
\dim(\star) &= 0 \\
\dim(s \to_A t) &= \dim(A) + 1 \\
\dim(\varnothing) &= -1 \\
\dim(\Gamma, (x:A)) &= \max(\dim(\Gamma), \dim(A))
\end{aligned}$$

And we define partial substitution functions for $\sigma\,\tau : \mathsf{Sub}, s\,t : \mathsf{Term}, A : \mathsf{Type}, x \in V$ as well as partial composition function given by:

$$\begin{aligned}
\star\,[\![\sigma]\!] &= \star \\
(s \to_A t)[\![\sigma]\!] &= s[\![\sigma]\!] \to_{A[\![\sigma]\!]} t[\![\sigma]\!] && \text{if this is defined} \\
\mathsf{coh}\ (\Gamma:A)[\tau][\![\sigma]\!] &= \mathsf{coh}\ (\Gamma:A)[\tau \circ \sigma] && \text{if } \tau \circ \sigma \text{ is defined} \\
x[\![\sigma]\!] &= t && \text{if } x \mapsto t \in \sigma \text{ and } x \text{ only occurs once in } \sigma \\
\langle\rangle \circ \sigma &= \langle\rangle \\
\langle\tau, x \mapsto t\rangle \circ \sigma &= \langle\tau \circ \sigma, x \mapsto t[\![\sigma]\!]\rangle && \text{if this is well defined}
\end{aligned}$$

2.2 Pasting Contexts and Globular Properties

Contexts in Catt correspond to finite computads (also known as finite polygraphs). Here we give rules to classify those contexts which form pasting diagrams.

$$\frac{}{x:\star \vdash_{pd} x:\star}\ \star \qquad\qquad \frac{\Gamma \vdash_{pd} x:\star}{\Gamma \vdash_{pd}}\ \checkmark$$

$$\frac{\Gamma \vdash_{pd} x:A \qquad y,f \in V\backslash\mathsf{FV}(\Gamma)}{\Gamma,(y:A),(f:x \to_A y) \vdash_{pd} f:x \to_A y}\ \Uparrow \qquad \frac{\Gamma \vdash_{pd} f:x \to_A y}{\Gamma \vdash_{pd} y:A}\ \Downarrow$$

Then given a pasting scheme Γ, the source $\partial^-(\Gamma)$ and target $\delta^+(\Gamma)$ can be defined:

$$\partial^-(x:\star) = \varnothing$$

$$\partial^-(\Gamma, y:A, f:x \to_A y) = \begin{cases} \partial^-(\Gamma), y:A, f:x \to_A y & \text{if } \dim(A)+1 < \dim(\Gamma) \\ \partial^-(\Gamma) & \text{if } \dim(A)+1 = \dim(\Gamma) \\ \Gamma & \text{if } \dim(A)+1 > \dim(\Gamma) \end{cases}$$

$$\mathsf{drop}(\Gamma, x:A) = \Gamma$$

$$\partial^+(x:\star) = \varnothing$$

$$\partial^+(\Gamma, y:A, f:x \to_A y) = \begin{cases} \partial^+(\Gamma), y:A, f:x \to_A y & \text{if } \dim(A)+1 < \dim(\Gamma) \\ \partial^+(\mathsf{drop}(\Gamma), y:A) & \text{if } \dim(A)+1 = \dim(\Gamma) \\ \mathsf{drop}(\Gamma), y:A & \text{if } \dim(A)+1 > \dim(\Gamma) \end{cases}$$

More generally we have the notion of a *globular* context, which is a context that does not contain any coherences in its syntax tree. These correspond to the computads which are just globular sets. Given an ambient context Γ, we can extend the dimension function to terms.

$$
\begin{aligned}
\dim_\Gamma(x) &= \dim(A) \quad \text{if } (x : A) \in \Gamma \\
\dim_\Gamma(\mathsf{coh}\ (\Delta : A)[\sigma]) &= \dim(A)
\end{aligned}
$$

Further for a term of dimension m, and $n \leqslant m$, we can define the nth source and target of the term as follows.

$$
\begin{aligned}
\delta^\epsilon_{\Gamma,n}(t) &= t && \text{if } n = \dim(x) \\
\delta^\epsilon_{\Gamma,n}(x) &= A^\epsilon_n && \text{if } (x : A) \in \Gamma \text{ and } n < \dim(x) \\
\delta^\epsilon_{\Gamma,n}(\mathsf{coh}\ (\Delta : A)[\sigma]) &= A^\epsilon_n[\![\sigma]\!] && \text{if } n < \dim(A)
\end{aligned}
$$

As before, the subscript context will be omitted when it can be inferred.

We also introduce a way to deconstruct types:

Definition 1. If A is a n dimensional type and $m < n$ then define A^ϵ_m as follows: First note that $m < n$ so n is at least 1 and so $A \equiv s \rightarrow_{A'} t$.

- If $m = n - 1$ then $A^-_m \equiv s$ and $A^+_m \equiv t$.

- Otherwise $A^\epsilon_m = (A')^\epsilon_m$.

Lastly we define unbiased compositions.

Definition 2. Mutually inductively define the *unbiased term* and *unbiased type* of a pasting diagram Γ, written $UTm(\Gamma)$ and $UTy(\Gamma)$.

- For a disk pasting diagram, the unbiased term is the top-dimensional variable, and the unbiased type is the type of this variable.

- For any other pasting diagram Γ, the unbiased type is given by $UTm(\partial^-(\Gamma)) \rightarrow UTm(\partial^+(\Gamma))$ and the unbiased term is given by $\mathsf{coh}\ (\Gamma : UTy(\Gamma))[\mathrm{id}]$

A coherence $\mathsf{coh}\ (\Delta : A)[\sigma]$ is *unbiased* if A is the unbiased type over Δ.

2.3 Properties of Syntax

We will now introduce some auxiliary definitions on the syntax of Catt. To do we want to consider an arbitrary type theory based on the syntax. This must contain a notion of definitional equality, and typing judgments.

Definition 3. A congruence $\sim$ is a tuple of equivalence relations on contexts, terms, substitutions and types. By abuse of notation we call all 4 of these relations $\sim$. We then must have the following rules:

$$\frac{\Gamma \sim \Gamma' \qquad A \sim A'}{\Gamma, (x : A) \sim \Gamma', (x : A')} \qquad\qquad \frac{\sigma \sim \sigma' \qquad t \sim t'}{\langle \sigma, x \mapsto t \rangle \sim \langle \sigma', x \mapsto t' \rangle}$$

$$\frac{A \sim A' \qquad s \sim s' \qquad t \sim t'}{s \to_A t \sim s' \to_{A'} t'} \qquad\qquad \frac{\Gamma \sim \Gamma' \qquad A \sim A' \qquad \sigma \sim \sigma'}{\mathsf{coh}\ (\Gamma : A)[\sigma] \sim \mathsf{coh}\ (\Gamma' : A')[\sigma']}$$

Example 1. Syntactic equality, $\equiv$, is a congruence.

We can now define the notion of a Catt-based type theory.

Definition 4. A Catt-*based type theory* consists of a congruence $=$, and typing judgments:

- $\Gamma \vdash$

- $\Gamma \vdash A$

- $\Gamma \vdash \sigma : \Delta$

- $\Gamma \vdash t : A$

for contexts Γ and Δ, type A, substitution σ, and term t. We further require that the following rules are satisfied:

$$\frac{\Gamma \vdash \qquad \Gamma = \Gamma'}{\Gamma' \vdash}$$

$$\frac{\Gamma \vdash A \qquad \Gamma = \Gamma' \qquad A = A'}{\Gamma' \vdash A}$$

$$\frac{\Gamma \vdash \sigma : \Delta \qquad \Gamma = \Gamma' \qquad \sigma = \sigma' \qquad \Delta = \Delta'}{\Gamma' \vdash \sigma' : \Delta'}$$

$$\frac{\Gamma \vdash t : A \qquad \Gamma = \Gamma' \qquad t = t' \qquad A = A'}{\Gamma' \vdash t' : A'}$$

2.4 Typing for Catt

To define the base type theory Catt, we need to define the definitional equality, and the typing judgments. The definitional equality will just be given by reflexivity. The typing rules for contexts, substitutions and types are given by:

$$\frac{}{\varnothing \vdash} \qquad\qquad \frac{\Gamma \vdash \qquad x \in V \backslash \mathsf{FV}(\Gamma) \qquad \Gamma \vdash A}{\Gamma, (x : A) \vdash}$$

$$\frac{\Gamma \vdash}{\Gamma \vdash \star} \qquad\qquad \frac{\Gamma \vdash A \qquad \Gamma \vdash s : A \qquad \Gamma \vdash t : A}{\Gamma \vdash s \to_A t}$$

$$\frac{\Delta \vdash}{\Delta \vdash \langle \rangle : \varnothing} \qquad \frac{\Delta \vdash \sigma : \Gamma \qquad \Gamma \vdash A \qquad \Delta \vdash t : A[\![\sigma]\!] \qquad x \in V \backslash \mathsf{FV}(\Gamma)}{\Delta \vdash \langle \sigma, x \mapsto t \rangle : \Gamma, (x : A)}$$

6

and then the rules for terms are given by:

$$\dfrac{\Gamma \vdash \quad (x : A) \in \Gamma}{\Gamma \vdash x : A} \; (\text{var})$$

$$\dfrac{\Gamma \vdash_{pd} \quad \Gamma \vdash s \rightarrow_A t \quad \Delta \vdash \sigma : \Gamma \quad \mathsf{supp}(s) = \partial^-(\Gamma) \quad \mathsf{supp}(t) = \partial^+(\Gamma)}{\Delta \vdash \mathsf{coh}\ (\Gamma : (s \rightarrow_A t))[\sigma] : (s \rightarrow_A t)[\![\sigma]\!]} \; (\text{comp})$$

$$\dfrac{\Gamma \vdash_{pd} \quad \Gamma \vdash A \quad \Delta \vdash \sigma : \Gamma \quad \mathsf{supp}(A) = \Gamma}{\Delta \vdash \mathsf{coh}\ (\Gamma : A)[\sigma] : A[\![\sigma]\!]} \; (\text{coh})$$

Theorem 5. *With a definitional equality given by syntactic equality,* Catt *forms a* Catt-*based type theory.*

3   Insertion

Now that we have defined the base type theory Catt, we are in a position to describe our insertion operation. We will first give an intuition to the action of this operation. Suppose we have the term $a \cdot (b \cdot c)$, where $a \cdot b$ represents (leaving out some implicit terms in the substitution):

$$\mathsf{coh}\ ((x : \star)(y : \star)(f : x \rightarrow y)(z : \star)(g : y \rightarrow z) : x \rightarrow z)[f \mapsto a, g \mapsto b]$$

In this case we are composing over some pasting diagram, and then one of the (locally maximal) arguments to this pasting diagram is itself a composition over a pasting diagram, in this case the variable g. Insertion then removes the variable g and everything in its support and "inserts" the inner pasting diagram in its place. This is demonstrated in Fig. 1 (where we have alpha renamed the inner pasting diagram to reduce confusion), where the final pasting diagram is simply giving the ternary composite, the unbiased version of the original composite.

To fully describe the new term, we also need two more constructions, as the coherence term constructor has three parts. Firstly we need a new substitution from the new pasting context to the ambient context, which assigns the arguments to the new composition. This can simply be done by mapping parts of the original pasting diagram (those in black in the figure) to their original arguments, and the parts of the inner pasting diagram (those in red in the figure) to the arguments of the inner composition. Secondly we need to construct the type part of the coherence (the A in $\mathsf{coh}\ (\Gamma : A)[\sigma]$). The most obvious way to do this is to transport the old type to the new pasting diagram, which will require a substitution from the old pasting diagram to the new inserted one.

Our insertion operation needs to trivialise associativity at all dimensions and all codimensions, and needs to consider composites of any shape so that the operation is natural enough to be confluent. Although the example we gave is relatively simple, describing the general operation at all dimensions can be tricky to reason about and so we will represent our pasting diagrams as labelled Batanin trees.

Definition 6. Given a set of variables L, a L-*labelled Batanin trees*, T can be defined inductively as a tuple (T_l, T_b) where T_l is a list of labels contained in L, T_b is a list of L labelled Batanin trees, and further the length of T_l is one greater than the length of T_b.

$$x \xrightarrow{f} y \xrightarrow{g} z \qquad\qquad \textcolor{red}{x'} \xrightarrow{\textcolor{red}{f'}} \textcolor{red}{y'} \xrightarrow{\textcolor{red}{g'}} \textcolor{red}{z'}$$

is sent to:

$$x \xrightarrow{f} \textcolor{red}{x'} \xrightarrow{\textcolor{red}{f'}} \textcolor{red}{y'} \xrightarrow{\textcolor{red}{g'}} \textcolor{red}{z'}$$

Figure 1: Insertion on $a \cdot (b \cdot c)$

Definition 7. From every labelled Batanin tree, we can obtain a context, and further show that this context is a pasting diagram. Given a tree T, we will denote the context $\lfloor T \rfloor$.

Lemma 8. *Given a context Γ, with a proof that it is a pasting diagram, we can form a labelled Batanin tree $\lceil \Gamma \rceil$. Further we have $\lfloor \lceil \Gamma \rceil \rfloor = \Gamma$ and $\lceil \lfloor T \rfloor \rceil = T$ for all pasting diagrams Γ and trees T.*

Proof. See [4] □

Definition 9. The linear height of a Batanin tree is the height of the tree before any branches occur. It can be defined recursively as:

$$\mathsf{lh}((T_l, T_b)) = \begin{cases} 1 + \mathsf{lh}(t) & \text{if } T_b = [t], \\ 0 & \text{otherwise} \end{cases}$$

Example 2. A tree is generated from a pasting context by having the labels be the 0-dimensional cells of the pasting diagram and then the tree between each 2 labels be the part of the pasting diagram suspended between those two labels. Consider these two examples, which we call Δ and Θ:

	Pasting diagram	Batanin tree
Δ		
Θ		

Here the first has a linear height of 0 and the second has a linear height of 1.

8

Definition 10. Each label in a Batanin tree can be associated with a path (a non empty list of natural numbers) specifying how to reach that label from the root of the tree. A path in a Batanin tree is *valid* if it is the path associated with one of the tree's labels.

Definition 11. Given a tree T and a locally maximal variable x, we can define the branching path of this variable, which we think of as the path to the last branching in the tree before reaching this variable, as follows: If T_b is empty then we return the path $[0]$. Otherwise T_b is not empty and as x is locally maximal, it must occur in the nth branch of T_b for some n. If this branch is linear then we return $[n]$, otherwise we append n to the branching path of x in this branch.

Definition 12 (Insertion)**.** Write $A^{[i..j]}$ for the list consisting of the ith element of A to the jth element of A (zero indexing). Suppose S and T are Batanin trees, and p is a valid path in S, and $\mathsf{lh}(T) \geqslant \mathsf{length}(p) - 1$. Then we define the insertion of T into S along path p, written $S \ll p\ T$, by recursion on p.

Suppose $p = [n]$, then let $(S \ll p\ T)_l$ be equal to $S_l^{[0..(n-1)]} +\!\!+ T_l +\!\!+ S_l^{[(n+2)..]}$ and $(S \ll p\ T)_b$ be equal to $S_b^{[0..(n-1)]} +\!\!+ T_b +\!\!+ S_b^{[(n+1)..]}$.

Suppose instead p has head n and non empty tail p'. By the linearity condition we have that $T_b = [T']$ for some t and let the nth element of S_b be S'. Then again let $(S \ll p\ T)_l = S_l^{[0..(n-1)]} +\!\!+ T_l +\!\!+ S_l^{[n+2..]}$ and in this case let $(S \ll p\ T)_b = S_b^{[0..(n-1)]} +\!\!+ [S \ll' p'T'] +\!\!+ S_b^{[(n+1)..]}$.

Finally for contexts Δ and Θ, and some locally maximal variable x of Δ, let $\Delta \ll x\ \Theta = |[\Delta] \ll p\ [\Theta]|$ where p is the branching path of x.

Definition 13. The internal substitution from $\iota_{\Delta,x,\Theta} : \Theta \to \Delta \ll x\ \Theta$ is given my mapping each variable to itself.

Definition 14. The *linear height* of a type A is the maximum n such that all terms contained in A with dimension less than or equal to A are variables.

Definition 15. Given a type A with $\Delta \vdash A$ with linear height at least one less than the length of the branching path length of x, we can define the external substitution $\kappa_{\Delta,x,\Theta,A} : \Delta \to \Delta \ll x\ \Theta$ by sending a variable y to y if variable y is in $\Delta \ll x\ \Theta$ and otherwise y is $\delta_n^\epsilon(x)$ for some $\epsilon \in \{+,-\}$ and n, and then we can map y to $\delta_n^\epsilon(\mathsf{coh}\ (\Theta : A)[\iota_{\Delta,x,\Theta}])$.

Example 3. Suppose we have contexts Δ and Θ from Example 2. The variable α has a branching height of 1 in Δ, and Θ has a linear height of 1, so the insertion $\Delta \ll \alpha\ \Theta$ is well defined and is given in Fig. 2 as a pasting diagram and as a tree. The internal substitution $\iota_{\Delta,\alpha,\Theta}$ maps Θ into the part of $\Delta \ll \alpha\ \Theta$ marked in red by mapping each variable to itself.

If we take A to be $f \cdot k \to h \cdot k$, then $\Delta \vdash A$ and A has linear height 1 and so we can define the external substitution $\kappa_{\Delta,\alpha,\Theta,A}$ which has the following action:

$$
\begin{aligned}
&\text{0-cells:} &&x \mapsto x' && y \mapsto y' && z \mapsto z \\
&\text{1-cells:} &&f \mapsto f' \quad g \mapsto h' \quad h \mapsto h \quad k \mapsto k \\
&\text{2-cells:} &&\alpha \mapsto \mathsf{coh}\ (\Theta : A)[\iota] \quad \beta \mapsto \beta
\end{aligned}
$$

Definition 16. Given $\sigma : \Delta \to \Gamma$, $\tau : \Theta \to \Gamma$, and variable x in S, we define the inserted substitution $\sigma \ll x\ \tau : \Delta \ll x\ \Theta \to \Gamma$, as sending any variable y originating from S to $\sigma(y)$ and any variable y originating from T to $\tau(y)$.
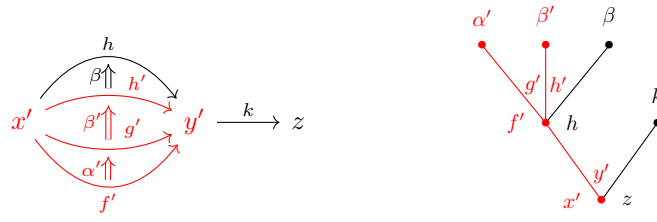
Figure 2: The insertion of Θ into Δ

Subscripts from the external and internal substitutions may be dropped when there is no ambiguity.

3.1 Catt$_{\mathsf{sa}}$

We define the type theory Catt$_{\mathsf{sa}}$ as having the same syntax as Catt. In addition to Catt it will have an extra definitional equality and modify the typing rules for terms. These will be defined simultaneously, though this will be well founded as the typing for a term will only depend on equality for terms of lower dimension and equality of terms will only depend on typing of terms of the same dimension.

The definitional equality will be given by the smallest congruence such that for all well typed terms $\mathsf{coh}\ (\Delta : A)[\sigma]$, locally maximal variable x of Δ with $\sigma(x)$ equal to an unbiased composite $\mathsf{coh}\ (\Theta : B)[\tau]$ with the branching height of x is less than or equal to the linear height of Θ we have:

$$\mathsf{coh}\ (\Delta : A)[\sigma] = \mathsf{coh}\ (\Delta \ll_x \Theta : A[\![\kappa_{\Delta,x,\Theta}]\!])[\sigma \ll_x \tau]$$

Explicitly, the definitional equality will have the following structural rules:

$$\frac{}{\varnothing = \varnothing} \qquad \frac{\Gamma = \Gamma' \qquad A = A'}{\Gamma, x : A = \Gamma', x : A'}$$

$$\frac{}{\star = \star} \qquad \frac{A = A' \qquad s = s' \qquad t = t'}{s \to_A t = s' \to_{A'} t'}$$

$$\frac{}{\langle\rangle = \langle\rangle} \qquad \frac{\sigma = \sigma' \qquad t = t'}{\langle\sigma, x \mapsto t\rangle = \langle\sigma', x \mapsto t'\rangle}$$

$$\frac{t = t'}{t' = t} \qquad \frac{t = t' \qquad t' = t''}{t = t''}$$

$$\frac{}{x = x} \qquad \frac{A = A' \qquad \sigma = \sigma'}{\mathsf{coh}\ (\Delta : A)[\sigma] = \mathsf{coh}\ (\Delta : A')[\sigma']}$$

Proposition 17. *This definitional equality is a congruence.*

We then modify the typing rules for terms to the following:

10

$$\frac{\Gamma \vdash \quad (x : A) \in \Gamma \quad A = B}{\Gamma \vdash x : B} \ (\mathrm{var}')$$

$$\frac{\begin{array}{ccc} \Gamma \vdash_{pd} & \Gamma \vdash s \to_A t & \Delta \vdash \sigma : \Gamma \\ \mathsf{supp}(s) = \partial^-(\Gamma) & \mathsf{supp}(t) = \partial^+(\Gamma) & B = (s \to_A t)[\![\sigma]\!] \end{array}}{\Delta \vdash \mathsf{coh} \ (\Gamma : (s \to_A t))[\sigma] : B} \ (\mathrm{comp}')$$

$$\frac{\Gamma \vdash_{pd} \quad \Gamma \vdash A \quad \Delta \vdash \sigma : \Gamma \quad \mathsf{supp}(A) = \Gamma \quad B = A[\![\sigma]\!]}{\Delta \vdash \mathsf{coh} \ (\Gamma : A)[\sigma] : B} \ (\mathrm{coh}')$$

From these rules we can deduce that the following conversion rule is admissable:

$$\frac{\Gamma \vdash t : A \quad A = B}{\Gamma \vdash t : A'}$$

Next, we will prove that the constructions make some form of sense, in that the constructed substitutions are indeed well typed in $\mathsf{Catt_{sa}}$. We start with the following definition.

Definition 18. A substitution σ from a globular set Γ to arbitrary context Δ *respects the globular conditions* if for all $x : s \to_A t$ in Γ, we have $\delta^-(\sigma(x)) = \sigma(s)$ and $\delta^+(\sigma(x)) = \sigma(t)$. Say a substitution is *well-formed* if it respects the globular conditions and maps each variable of Γ to a term which is well typed in Δ.

Lemma 19. *If* $\Delta \vdash u : A$ *then* $\delta_n^\epsilon(u) = A_n^\epsilon$ *for all* $n < \dim(A)$.

Proof. Proceed by induction on the proof of $\Delta \vdash u : A$.

- If it is proved using a variable rule, then the result follows straight from definition of δ and compatibility of $(-)_n^\epsilon$ and equality.

- If it is proved with the coherence or comp rule then $u \equiv \mathsf{coh} \ (\Gamma : B)[\sigma]$ and $A = B[\![\sigma]\!]$. Then $\delta_n^\epsilon(u) \equiv B_n^\epsilon[\![\sigma]\!] \equiv (B[\![\sigma]\!])_n^\epsilon = A_n^\epsilon$.

$\square$

Theorem 20. *In* $\mathsf{Catt_{sa}}$, *if* Γ *is a globular set then* $\Delta \vdash \sigma : \Gamma$ *if and only if* σ *is well-formed.*

Proof. Proceed by induction on the length of σ. If σ is empty, then it is both well-formed and well-typed.

Now consider $\langle \sigma, x \mapsto u \rangle$ and first suppose $\Delta \vdash \langle \sigma, x \mapsto u \rangle : \Gamma, x : A$. By inductive hypothesis, σ is well-formed. In particular each variable of Γ is sent by σ to a well-typed term of Δ. Further, by assumption $\Delta \vdash u : A[\![\sigma]\!]$ and so x is also sent to a well-typed term. Now we need to show this substitution respects the globular conditions. The only case that does not immediately follow from inductive hypothesis is the one for x. If $A \equiv \star$ then there is nothing to do. Otherwise suppose $A \equiv s \to_{A'} t$. Then $\Delta \vdash u : s[\![\sigma]\!] \to_{A'[\![\sigma]\!]} t[\![\sigma]\!]$. Then by Lemma 19 we have $\delta^-(u) = s[\![\sigma]\!]$ and $\delta^+(u) = t[\![\sigma]\!]$ as required.

Now suppose $\langle \sigma, x \mapsto u \rangle$ is well-formed. As a well-formed substitution maps all the variables in the source context, we must have $x : A \in \Gamma$. Then we have that $\Delta \vdash u : B$ for some type B by the substitution being well-formed. If $B \equiv \star$ then A must also be $\star$ so we are done. Otherwise, for all ϵ and n, we have that $\delta_n^\epsilon(u) = B_n^\epsilon$ by Lemma 19 and $\delta_n^\epsilon(u) = A_n^\epsilon[\![\sigma]\!]$ by σ being well formed. Therefore $A[\![\sigma]\!] = B$ and so $\Delta \vdash u : A[\![\sigma]\!]$ by an application of the conversion rule. $\square$

We now introduce a further equality operation, which effectively says that two terms are definitionally equal but also syntactically equal on any part of the term which has suitably high dimension. This will be useful later for proving confluence.

Definition 21. Define the equality relation $=_n$ on terms, types, and substitutions by the following rules.

- If s and t are terms with $\dim(s) < n$ and $\dim(t) < n$ then $s =_n t$ if and only if $s = t$.

- If s is a variable of dimension greater than or equal to n then $s =_n t$ if and only if $s \equiv t$.

- If $s \equiv \mathsf{coh}\ (\Delta : A)[\sigma]$ with $\dim(s) \geqslant n$ then $s =_n t$ if and only if $t \equiv \mathsf{coh}\ (\Delta : A')[\sigma']$ and $A =_n A'$ and $\sigma =_n \sigma'$.

- For types $\star =_n \star$ and $s \to_A t =_n s' \to_{A'} t'$ if and only if $s =_n s'$, $t =_n t'$, and $A =_n A'$.

- For substitutions $\langle t_0, \ldots, t_k \rangle =_n \langle s_0, \ldots, s_k \rangle$ if and only if $t_i =_n s_i$ for all i.

Next fix pasting contexts Δ and Θ, a variable x in Δ, and a type A with $\Theta \vdash A$, such that the linear height of A is at least one less than the branching height of x.

Proposition 22. $\iota : \Theta \to \Delta \lll x\ \Theta$ *is well-formed.*

Proof. As all the variables in Θ have the same type in the inserted context, this is clearly a well-formed substitution. $\qquad\square$

Proposition 23. *The substitution* $\kappa : \Delta \to \Delta \lll x\ \Theta$ *is well-formed.*

Proof. For this to be a well-formed substitution, the only interesting case is where y is not in the support of x, yet one or both of it's source and target variables are. Suppose variable z is the target of y and the nth source of x in Δ. An inspection of the algorithm shows that the target of y in $\Delta \lll x\ \Theta$ is the unique variable in the nth source of Θ, which we call z'. However now we have that $\kappa(z) \equiv \delta_n^-(\mathsf{coh}\ (\Theta : A)[\iota]) \equiv z' \equiv \delta^-(y)$ and so $\kappa(\delta_\Delta^-(y)) \equiv \delta_{\Delta \lll \Theta}^-(\kappa(y))$. The case where z was the source of y follows similarly. $\qquad\square$

Now further assume that we have well-formed $\sigma : \Delta \to \Gamma$ and $\tau : \Theta \to \Gamma$ with $\sigma(x) = \mathsf{coh}\ (\Theta : A)[\tau]$.

Proposition 24. $\tau = (\sigma \lll x\ \tau) \circ \iota$ *and* $\sigma = (\sigma \lll x\ \tau) \circ \kappa$.

Proof. The first equality follows directly from the definition of $\sigma \lll x\ \tau$. To prove the second, it suffices to prove the cases for y in the support of x, as the rest hold definitionally. If $y = \delta_n^\epsilon(x)$, then $\kappa(y) = \delta_n^\epsilon(\mathsf{coh}\ (\Theta : A)[\iota])$. Further we know that, as $\sigma(x) = \mathsf{coh}\ (\Theta : A)[\tau]$ and as all the substitutions are well-formed, $y[\![\sigma]\!]$ is equal to $\delta_n^\epsilon(\mathsf{coh}\ (\Theta : A)[\tau])$. Therefore:

$$
\begin{aligned}
y[\![\sigma]\!] &= \delta_n^\epsilon(\mathsf{coh}\ (\Theta : A)[\tau]) \\
&= \delta_n^\epsilon(\mathsf{coh}\ (\Theta : A)[\mathrm{id}])[\![\tau]\!] \\
&= \delta_n^\epsilon(\mathsf{coh}\ (\Theta : A)[\mathrm{id}])[\![(\sigma \lll x\ \tau) \circ \iota]\!] \\
&= (\delta_n^\epsilon(\mathsf{coh}\ (\Theta : A)[\mathrm{id}])[\![\iota]\!])[\![\sigma \lll x\ \tau]\!] \\
&= (\delta_n^\epsilon(\mathsf{coh}\ (\Theta : A)[\iota]))[\![\sigma \lll x\ \tau]\!] \\
&= (y[\![\kappa]\!])[\![\sigma \lll x\ \tau]\!] \\
&= y[\![(\sigma \lll x\ \tau) \circ \kappa]\!]
\end{aligned}
$$

and so we get the required syntactic equality. □

Proposition 25. $\sigma \ll x \ \tau$ *is well-formed.*

Proof. To check that $\sigma \ll x \ \tau$ is well-formed, we again look at the one problematic case where y originates from Δ yet its (without loss of generality) target z originates from Θ. Then we need to show that $\delta^+(\sigma(y)) = \tau(z)$. But then as κ is a well-formed substitution we have that there is $z' = \delta_\Delta^+(y)$ with $\kappa(z') = z$. As σ is well-formed we have that $\delta^+(\sigma(y)) = \sigma(z') = (\kappa(z'))[\![\sigma \ll x \ \tau]\!] = (\sigma \ll x \ \tau)(z) = \tau(z)$ as required. □

3.2 Universal Property of Insertion

We first must introduce some theory about disk contexts:

Definition 26. Recursively define the disk contexts D_n as D_0 being the context $d_0^- : \star$ and D_{n+1} being the context $D_n, d_n^+ : d_{n-1}^- \to d_{n-1}^+, d_{n+1}^- : d_n^- \to d_n^+$.

Note that each disk context has a single maximal dimension cell, and any substitution out of a disk context is fully determined by where this cell is sent.

Definition 27. A substitution $D_n \to \Gamma$ is fully determined by its action on d_n^-. If $\Gamma \vdash t$ and t is n-dimensional, then let $\bar{t} : D_n \to \Gamma$ be the substitution that sends d_m^ϵ to $\delta_m^\epsilon(t)$ (and so in particular sends d_n^- to t).

Given pasting contexts Δ and Θ, a variable x in Δ and a type A with $\Theta \vdash A$, we can classify insertion by a universal property.

Definition 28. When $\dim(A) = \dim(x) = n$, we say that an insertion of (Θ, A) into Δ via x is the following pushout:

$$
\begin{array}{ccc}
D_n & \xrightarrow{\ \overline{x}\ } & \Delta \\
{\scriptsize \overline{\mathsf{coh}\ (\Theta:A)[\mathrm{id}]}} \downarrow & \llcorner & \downarrow \\
\Theta & \longrightarrow & I
\end{array}
$$

Note that as a pushout, this notion is unique up to isomorphism. If we further require that the insertion is a pasting diagram, then (if it exists) this uniquely specifies this colimit.

Theorem 29. *When* $\dim(A) = \dim(x) = n$*, x is locally maximal, and A is linear up to at least one less than the length of the branching path of x, then the following is a pushout in* $\mathsf{Catt_{sa}}$*:*

$$
\begin{array}{ccc}
D_n & \xrightarrow{\ \overline{x}\ } & \Delta \\
{\scriptsize \overline{\mathsf{coh}\ (\Theta:A)[\mathrm{id}]}} \downarrow & {\scriptstyle \kappa} \downarrow & \\
\Theta & \xrightarrow{\ \iota\ } & \Delta \ll x \ \Theta
\end{array}
$$

and hence $\Delta \ll x \ \Theta$ is an insertion of (Θ, A) into Δ along x. Further it is a pasting diagram and so is the unique pasting diagram with this property.

Proof. We have already checked that all these contexts and substitutions are well typed. To check commutativity of the square it suffices to check that the maximal variable is sent to the same term

13

by each substitution. Along the top it is sent to $x[\![\kappa]\!]$, which is defined to be $\mathsf{coh}\ (\Theta : A)[\iota] = (\mathsf{coh}\ (\Theta : A)[\mathrm{id}])[\![\iota]\!]$, which is the term it is sent to along the bottom.

Now suppose that we have Γ, σ, and τ such that the following commutes:

Then we need to generate a substitution $\Delta \ll x\ \Theta \to \Gamma$, for which we can give $\sigma \ll x\ \tau$. Any variable l that originates from Δ must be sent to $\sigma(l)$ and any variable l that originates from Θ must be sent to $\tau(l)$. This ensures uniqueness. The commutativity conditions are given in Proposition 24. $\qquad\square$

To finish this section we note that the conditions we have required for insertion to work are much weaker than what we have used for the definition of $\mathsf{Catt_{sa}}$. Firstly, for the equality in $\mathsf{Catt_{sa}}$, we require that the inner term be fully unbiased instead of its type being linear up to a certain height, and further the insertion operation does not depend on the inner term being a composition instead of a coherence, which we explore in the next example.

Example 4. Suppose we have pasting diagram Δ with locally maximal x, and $\sigma : \Delta \to \Gamma$ with $\sigma(x) \equiv \mathrm{id}(t)$. Then the pasting diagram of $\mathrm{id}(t)$ is D_n for some n. As D_n is fully linear, it satisfies the conditions for insertion, and so we can form the insertion $\Delta \ll x\ D_n$ and from a term $\mathsf{coh}\ (\Delta : A)[\sigma]$, we could form the term $\mathsf{coh}\ (\Delta \ll x\ D_n : A[\![\kappa]\!])[\sigma \ll x\ \bar{t}]$. This term has the same form[1] as the result of applying the pruning operation from [5] to the original term.

That pruning can be realised as a case of insertion has two exciting consequences. Firstly, this means that we can also realise pruning as a universal property, which was not previously known and allows some proofs to be written in a neat way. It also suggests that the two operations should be compatible and it is possible that they can be combined to get a strictly associative and unital type theory. Proving confluence of such a system is ongoing work.

3.3 Properties of equality

We can now prove some facts about the equality of $\mathsf{Catt_{sa}}$.

Lemma 30. *Suppose there exists an insertion of (Θ, A) into Δ and we have substitutions $\rho : \Delta \to \Gamma$, $\tau : \Theta \to \Gamma$ and $\sigma : \Gamma \to \Gamma'$ with $\rho(x) = \mathsf{coh}\ (\Theta : A)[\tau]$. Then $\sigma \circ (\rho \ll x\ \tau) = (\sigma \circ \rho) \ll x\ (\sigma \circ \tau)$.*

Proof. Take $y \in \mathsf{Var}(\Delta \ll x\ \Theta)$. If $y \in \mathsf{Var}(\Theta)$ then:

$$
\begin{aligned}
(\sigma \circ (\rho \ll x\ \tau))(y) &= ((\rho \ll x\ \tau)(y))[\![\sigma]\!] \\
&= (\tau(y))[\![\sigma]\!] \\
&= (\sigma \circ \tau)(y) \\
&= ((\sigma \circ \rho) \ll x\ \sigma \circ \tau)(y)
\end{aligned}
$$

---

[1]The two operations are not identical, as inserting an identity takes as many arguments from the inner term as possible, and pruning discards these arguments, taking them from the outer term instead. However, the normal forms of these two operations (with appropriate structural rules) should be identical.

If $y \in \mathsf{Var}(\Delta)$ then:

$$\begin{aligned}
(\sigma \circ (\rho \lll x\ \tau))(y) &= ((\rho \lll x\ \tau)(y))[\![\sigma]\!] \\
&= (\rho(y))[\![\sigma]\!] \\
&= (\sigma \circ \rho)(y) \\
&= ((\sigma \circ \rho) \lll x\ \sigma \circ \tau)(y)
\end{aligned}$$

As these are the only two cases, this completes the proof. $\qquad\square$

Lemma 31. *For all well typed $\sigma = \sigma'$ we have:*

- $t[\![\sigma]\!] = t'[\![\sigma']\!]$ *for all well typed terms $t = t'$*

- $A[\![\sigma]\!] = A'[\![\sigma']\!]$ *for all well typed types $A = A'$*

- $\sigma \circ \tau = \sigma \circ \tau'$ *for all well typed substitutions $\tau = \tau'$.*

Proof. Prove by mutual induction on the equality of types, terms, and substitutions.

- Take equality $A = A'$. If this is an instance of $\star = \star$ then $\star[\![\rho]\!] \equiv \star$ for any ρ, so the result is trivial. If it is instead an instance of $s \to_B t = s' \to_{B'} t'$ from $s = s'$, $t = t'$, and $B = B'$. Then by inductive hypothesis we get:

$$\begin{aligned}
A[\![\sigma]\!] &= (s \to_B t)[\![\sigma]\!] \\
&= s[\![\sigma]\!] \to_{B[\![\sigma]\!]} t[\![\sigma]\!] \\
&= s'[\![\sigma']\!] \to_{B'[\![\sigma']\!]} t'[\![\sigma']\!] \\
&= s' \to_{B'} t'[\![\sigma']\!] \\
&= A'[\![\sigma']\!]
\end{aligned}$$

- Take equality $t = t'$. If this is an instance of $x = x$ then we need that $\sigma(x) = \sigma'(x)$, which follows from $\sigma = \sigma'$. If it is an instance of transitivity or symmetry, then we can simply use inductive hypothesis and transitivity or symmetry on the resulting terms. For the structural coherence case we have $\mathsf{coh}\ (\Delta : B)[\rho] = \mathsf{coh}\ (\Delta : B')[\rho']$ with $B = B'$ and $\rho = \rho'$. Then $t[\![\sigma]\!] \equiv \mathsf{coh}\ (\Delta : B)[\sigma \circ \rho]$ and $t'[\![\sigma']\!] \equiv \mathsf{coh}\ (\Delta : B')[\sigma' \circ \rho']$. Using inductive hypothesis we get that $\sigma \circ \rho = \sigma' \circ \rho'$ and so we can apply the structural coherence rule to get the required equality.

  This leaves the insertion case. Suppose we have $\mathsf{coh}\ (\Delta : B)[\rho]$ with $\rho(x) = \mathsf{coh}\ (\Theta : C)[\tau]$ and the equality is $\mathsf{coh}\ (\Delta : B)[\rho] = \mathsf{coh}\ (\Delta \lll x\ \Theta : A[\![\kappa_{\Delta,x,\Theta}]\!])[\rho \lll x\ \tau]$. When we apply the substitutions we get that $t[\![\sigma]\!] \equiv \mathsf{coh}\ (\Delta : B)[\sigma \circ \rho]$ and $t'[\![\sigma']\!] \equiv \mathsf{coh}\ (\Delta \lll x\ \Theta : B[\![\kappa_{\Delta,x,\Theta}]\!])[\sigma' \circ (\rho \lll x\ \tau)]$. As $(\sigma \circ \rho)(x) \equiv \mathsf{coh}\ (\Theta : C)[\sigma \circ \tau]$, we can certainly apply an insertion to $t[\![\sigma]\!]$ to get $\mathsf{coh}\ (\Delta \lll x\ \Theta : B[\![\kappa_{\Delta,x,\Theta}]\!])[(\sigma \circ \rho) \lll x\ (\sigma \circ \tau)]$.

  It remains to show modulo structural rules that $\sigma' \circ (\rho \lll x\ \tau) = (\sigma \circ \rho) \lll x\ (\sigma \circ \tau)$. By inductive hypothesis we have that $\sigma' \circ (\rho \lll x\ \tau) = \sigma \circ (\rho \lll x\ \tau)$, and the remaining equality is given by Lemma 30.

- The case for substitutions follows easily from inductive hypothesis.

$\qquad\square$

Lemma 32. *The support operation is invariant under equality.*

Proposition 33. $\mathsf{Catt_{sa}}$ *is a* Catt-*based type theory.*

Proof. We proceed by mutually proving the following statements by induction on the equality proofs.

- For any $\Gamma = \Gamma'$ we have $\Gamma \vdash$ if and only if $\Gamma' \vdash$.

- For any $A = A'$ we have that for any $\Gamma = \Gamma'$ that $\Gamma \vdash A$ if and only if $\Gamma' \vdash A'$.

- For any $\sigma = \sigma'$, for all $\Gamma = \Gamma'$ and $\Delta = \Delta'$, $\Gamma \vdash \sigma : \Delta$ if and only if $\Gamma' \vdash \sigma' : \Delta'$.

- For any $t = t'$, for all $\Gamma = \Gamma'$ and $A = A'$, $\Gamma \vdash t : A$ if and only if $\Gamma' \vdash t' : A'$.

Now we split into cases:

- Suppose we have $\Gamma = \Gamma'$ and want to prove that $\Gamma \vdash$ if and only if $\Gamma' \vdash$. Case splitting on the equality:

    - If $\Gamma = \Gamma'$ is an instance of $\varnothing = \varnothing$ then the result is trivial.
    - Otherwise we have $\Delta, x : A = \Delta, x : A'$ with $\Delta = \Delta'$ and $A = A'$. If $\Delta, x : A \vdash$ then $\Delta \vdash$ and $\Delta \vdash A$. By inductive hypothesis, we have that $\Delta' \vdash$ and $\Delta' \vdash A'$ and so $\Delta', x : A' \vdash$. The other direction is symmetrical.

- Suppose we have $A = A'$. We want to show for all $\Gamma = \Gamma'$ that $\Gamma \vdash A$ if and only if $\Gamma' \vdash A'$. Case splitting on $A = A'$.

    - If it is a case of $\star = \star$ then the result is trivially true.
    - Otherwise the equality is an instance of $s \to_A t = s' \to_{A'} t'$ with $s = s'$, $A = A'$, $t = t'$. As above, the result follows easily from inductive hypothesis.

- Suppose we have $\sigma = \sigma'$. We want to show for all $\Gamma = \Gamma'$ and $\Delta = \Delta'$ that $\Gamma \vdash \sigma : \Delta$ if and only if $\Gamma' \vdash \sigma' : \Delta'$: Cases splitting on $\sigma = \sigma'$:

    - If the equality is an instance of $\langle\rangle = \langle\rangle$ then again the result is trivial.
    - If the equality is an instance of $\langle \tau, x \mapsto t \rangle = \langle \tau, x \mapsto t' \rangle$ with $\tau = \tau'$ and $t = t'$ and then suppose $\Gamma \vdash \sigma : \Delta$. Then it must be that $\Delta \equiv \Theta, x : A$ and so as $\Delta = \Delta'$, $\Delta' \equiv \Theta', x : A'$ with $\Theta = \Theta'$ and $A = A'$. We further have that $\Gamma \vdash \tau : \Theta$ and $\Gamma \vdash t : A[\![\tau]\!]$. By Lemma 31, $A[\![\tau]\!] = A'[\![\tau']\!]$ and so by inductive hypothesis we have that $\Gamma' \vdash \tau' : \Theta'$ and $\Gamma' \vdash t' : A'[\![\tau']\!]$ and so $\Gamma' \vdash \sigma' : \Delta'$ as required. The other direction again follows the same argument.

- Suppose we have $t = t'$. We want to show for all $\Gamma = \Gamma'$ and $A = A'$ that $\Gamma \vdash t : A$ if and only if $\Gamma' \vdash t' : A'$. Case split on $t = t'$:

    - If it is a case of $x = x$ then suppose we have $\Gamma \vdash x : A$. Then this typing rule must be derived from (var') so $x : B \in \Gamma$ and $A = B$. Then $x : B' \in \Gamma$ with $B = B'$ as $\Gamma = \Gamma'$ and so as equality is a congruence we have $A' = A = B = B'$ and so we can apply the (var') rule to get $\Gamma' \vdash x : A'$. The other direction is symmetrical.

16

- If it is a case of symmetry, then the result follows directly from inductive hypothesis.

- If it is a case of transitivity, then the result also follows directly from inductive hypothesis.

- If it is from the structural rule on coherences we have $\mathsf{coh}\ (\Delta : B)[\sigma] = \mathsf{coh}\ (\Delta : B')[\sigma']$ with $B = B'$ and $\sigma = \sigma'$. Suppose we have $\Gamma \vdash \mathsf{coh}\ (\Delta : B)[\sigma] : A$ by the composition rule or coherence rule. In this case we have $A = B[\![\sigma]\!]$, $\Delta \vdash B$, and $\Gamma \vdash \sigma : \Delta$. Then by inductive hypothesis we have $\Delta \vdash B'$ and $\Gamma' \vdash \sigma' : \Delta$. Further we have $A' = A = B[\![\sigma]\!] = B'[\![\sigma']\!]$ by equality being a congruence and Lemma 31. Then, using Lemma 32 and a coherence or composition rule, we get that $\Gamma' \vdash t' : A'$. The other direction follows similarly.

- The last case is where the equality is an instance of insertion. Here we have $t \equiv \mathsf{coh}\ (\Delta : B)[\sigma]$ with $\sigma(x) = \mathsf{coh}\ (\Theta : C)[\tau]$ and $t' \equiv \mathsf{coh}\ (\Delta \ll x\ \Theta : B[\![\kappa]\!])[\sigma \ll x\ \tau]$.

   For the reverse direction we assume that $\Gamma \vdash t' : A'$. By assumption of insertion we have that $\Gamma \vdash t : D$ for some type D. Then by inspection of rules we must have that $D = B[\![\sigma]\!]$. Similarly we have that $A' = B[\![\kappa]\!][\![\sigma \ll x\ \tau]\!] = B[\![(\sigma \ll x\ \tau) \circ \kappa]\!] = B[\![\sigma]\!]$ and so $D = A'$.

   Conversely assume that $\Gamma \vdash t : A$. Using inductive hypothesis we get that B is well typed in Δ and σ is well typed from Δ to Γ'. Then by results in the previous section we have that $\Delta \ll x\ \Theta$ is a pasting context, κ is well typed and so $B[\![\kappa]\!]$ is well typed, and $\sigma \ll x\ \tau$ is a valid substitution from $\Delta \ll x\ \Theta$ to Γ'. Therefore t' can be typed with any type equal to $B[\![\kappa]\!][\![\sigma \ll x\ \tau]\!] = B[\![\sigma]\!]$. Now suppose $A' = A$. Similarly to before we have $A = B[\![\sigma]\!]$ so by $=$ being a congruence we have $A' = B[\![\sigma]\!]$ and so $\Gamma' \vdash t' : A'$.

This completes all cases and is sufficient to show what was required. $\qquad\square$

4 Reduction

So far we have given a type theory which models strictly associative infinity categories. Unfortunately, our type theory currently has no way to decide equality and by extension perform type checking. To solve this we introduce a reduction relation, show that it agrees with the definitional equality, and show that it is complete by showing it has strong normalisation and local confluence.

Definition 34 (Reduction on types)**.** The following rules are admissible for reducing types.

- If $s \rightsquigarrow s'$:
$$s \rightarrow_A t \rightsquigarrow s' \rightarrow_A t$$

- If $t \rightsquigarrow t'$:
$$s \rightarrow_A t \rightsquigarrow s \rightarrow_A t'$$

- If $A \rightsquigarrow A'$:
$$s \rightarrow_A t \rightsquigarrow s \rightarrow_{A'} t$$

Definition 35 (Reduction on substitutions)**.** The following rules can be used to reduce substitutions:

- If $\sigma \rightsquigarrow \sigma'$: $\langle \sigma, t \rangle \rightsquigarrow \langle \sigma', t \rangle$

- If $t \rightsquigarrow t'$: $\langle \sigma, t \rangle \rightsquigarrow \langle \sigma, t' \rangle$

This effectively allows any term in the substitution to be reduced.

Definition 36 (Reduction on terms). The following rules can be used to reduce terms:

- (Insertion): If x is locally maximal in Δ and $\sigma(x)$ is the unbiased composite $\mathsf{coh}\ (\Theta : B)[\tau]$ with the branching height of x being less than or equal to the linear height of Θ then:

$$\mathsf{coh}\ (\Delta : A)[\sigma] \rightsquigarrow \mathsf{coh}\ (\Delta \ll x\ \Theta : A[\![\kappa_{\Delta,x,\Theta}]\!])[\sigma \ll x\ \tau]$$

- (Cell reduction): The type of a coherence can be reduced, so if $A \rightsquigarrow A'$ then:

$$\mathsf{coh}\ (\Delta : A)[\sigma] \rightsquigarrow \mathsf{coh}\ (\Delta : A')[\sigma]$$

- (Argument reduction): The arguments to a coherence can be reduced, so if $\sigma \rightsquigarrow \sigma'$ then:

$$\mathsf{coh}\ (\Delta : A)[\sigma] \rightsquigarrow \mathsf{coh}\ (\Delta : A)[\sigma']$$

Lemma 37. *For all terms $t \rightsquigarrow t'$, types $A \rightsquigarrow A'$, and substitutions $\sigma \rightsquigarrow \sigma'$, and another substitution τ, then $t[\![\tau]\!] \rightsquigarrow t'[\![\tau]\!]$, $A[\![\tau]\!] \rightsquigarrow A'[\![\tau]\!]$ and $\tau \circ \sigma \rightsquigarrow \tau \circ \sigma'$.*

Proof. Proceed by mutual induction on t, A, and σ.

- For substitutions $\sigma = \langle t_0, \ldots, t_n \rangle$, we have $\sigma \rightsquigarrow \sigma'$ and therefore there must be some i with $t_i \rightsquigarrow t'_i$ and $\sigma' = \langle t_0, \ldots, t'_i, \ldots, t_n \rangle$. Further, $\tau \circ \sigma = \langle t_0[\![\tau]\!], \ldots, t_n[\![\tau]\!] \rangle$ and $\tau \circ \sigma' = \langle t_0[\![\tau]\!], \ldots, t'_i[\![\tau]\!], \ldots, t_n[\![\tau]\!] \rangle$ and so by inductive hypothesis $t_i[\![\tau]\!] \rightsquigarrow t'_i[\![\tau]\!]$ and so $\tau \circ \sigma \rightsquigarrow \tau \circ \sigma'$.

- For types we get a proof similar to above where we apply the inductive hypothesis to some subterm.

- For terms t, we note that t cannot be a variable as variables do not reduce. Therefore $t \equiv \mathsf{coh}\ (\Delta : A)[\sigma]$ and $t[\![\tau]\!] \equiv \mathsf{coh}\ (\Delta : A)[\tau \circ \sigma]$. Now we case split on the possible reductions of t.

  - Cell reduction: The type A is unaffected by substitution so this remains unchanged.
  - Argument reduction: Follows from inductive hypothesis.
  - Insertion: Suppose $t \rightsquigarrow t'$ is an insertion along x and $\sigma(x) \equiv \mathsf{coh}\ (\Theta : B)[\rho]$. Then $(\tau \circ \sigma)(x) \equiv \mathsf{coh}\ (\Theta : B)[\tau \circ \rho]$ and so $t[\![\tau]\!]$ admits an insertion. We must have $t' \equiv \mathsf{coh}\ (\Delta \ll x\ \Theta : B[\![\kappa]\!])[\sigma \ll x\ \rho]$ and $t[\![\tau]\!]$ reduces by insertion to $\mathsf{coh}\ (\Delta \ll x\ \Theta : B[\![\kappa]\!])[(\tau \circ \sigma) \ll x\ (\tau \circ \rho)]$. It remains to show that $\tau \circ (\sigma \ll x\ \rho) \equiv (\tau \circ \sigma) \ll x\ (\tau \circ \rho)$ but this is given by Lemma 30.

$\square$

We end this section by showing that reduction agrees with equality in the following way.

Definition 38. Let $\rightsquigarrow'$ be the reduction $\rightsquigarrow$ restricted to terms that are well typed, and $\rightsquigarrow'_{\mathsf{rts}}$ be the reflexive transitive symmetric closure of this reduction.

Proposition 39. *On valid terms, types and substitutions, the equality relation $=$ agrees with $\leadsto'_{rts}$.*

Proof. It is clear that if $p \leadsto'_{rts} q$ then $p = q$. Therefore we only need to prove the other direction. As in [5], we define a term p to be *conservative* if for all q with $p = q$, we have $p \leadsto'_{rts} q$. The only case that doesn't follow almost immediately from inductive hypothesis is the case where $p = q$ is by insertion. Therefore suppose $p \equiv \mathsf{coh}\ (\Delta : A)[\sigma]$ and $\sigma(x) = \mathsf{coh}\ (\Theta : U)[\tau]$ and $q \equiv \mathsf{coh}\ (\Delta \ll x\ \Theta : A[\![\iota]\!])[\sigma \ll x\ \tau]$. Then by induction on subterms we have that $\sigma(x) \leadsto'_{rts} \mathsf{coh}\ (\Theta : U)[\tau]$ and so $p \leadsto'_{rts} \mathsf{coh}\ (\Delta : A)[\sigma'] \leadsto' q$ where σ' is σ with $\sigma(x)$ replaced with $\mathsf{coh}\ (\Theta : U)[\tau]$. $\qquad\square$

4.1 Termination

In this section we prove termination of the reduction scheme presented above. To do this we will map each term to an ordinal, show that all reductions reduce this ordinal.

Definition 40. The natural sum of ordinals $\alpha \mathbin{\#} \beta$ is defined by simultaneous induction on α and β by:

$$0 \mathbin{\#} 0 = 0$$
$$\alpha \mathbin{\#} \beta = \sup_{\substack{\alpha' < \alpha \\ \beta' < \beta}} \{S(\alpha \mathbin{\#} \beta'), S(\alpha' \mathbin{\#} \beta)\}$$

Lemma 41. *The natural sum of ordinals is associative, commutative, and strictly monotone in both arguments.*

Proof. See [11]. $\qquad\square$

Definition 42. Define the syntactic depth sd of substitutions, typed, and terms by mutual induction:

- For substitutions we have:
$$\mathsf{sd}(\langle t_0, \ldots, t_n \rangle) = \overset{n}{\underset{i=0}{\#}} t_i$$

- For types we have $\mathsf{sd}(\star) = 0$ and:
$$\mathsf{sd}(s \rightarrow_A t) = \mathsf{sd}(s) \mathbin{\#} \mathsf{sd}(A) \mathbin{\#} \mathsf{sd}(t)$$

- For variables x, $\mathsf{sd}(x) = 0$.

- For coherences define:
$$\mathsf{sd}(\mathsf{coh}\ (\Delta : A)[\sigma]) = \omega^{\dim(A)} \mathbin{\#} \mathsf{sd}(A) \mathbin{\#} \mathsf{sd}(\sigma)$$

Lemma 43. *For terms $t \leadsto t'$, $\mathsf{sd}(t) > \mathsf{sd}(t')$, for types $A \leadsto A'$, $\mathsf{sd}(A) > \mathsf{sd}(A')$, and for substitutions $\sigma \leadsto \sigma'$, $\mathsf{sd}(\sigma) > \mathsf{sd}(\sigma')$.*

Proof. By mutual induction on term reductions $t \leadsto t'$, type reductions $A \leadsto A'$ and substitution reductions $\sigma \leadsto \sigma'$:

- For substitutions, if $\sigma \rightsquigarrow \sigma'$ then suppose $\sigma = \langle t_0, \ldots, t_n \rangle$ and $\sigma' = \langle t'_0, \ldots, t'_n \rangle$. Then for some i we must have $t_i \rightsquigarrow t'_i$ and $t_k = t'_k$ for $k \neq i$. By inductive hypothesis $\mathsf{sd}(t_i) > \mathsf{sd}(t'_i)$ and so $\mathsf{sd}(\sigma) > \mathsf{sd}(\sigma')$.

- For types, $\star$ admits no reductions so suppose $A \equiv s \rightarrow_B t$. If $A' \equiv s' \rightarrow_B t$ where $s \rightsquigarrow s'$ then by inductive hypothesis $\mathsf{sd}(s) > \mathsf{sd}(s')$ and so $\mathsf{sd}(A) > \mathsf{sd}(A')$. The cases where $A' \equiv s \rightarrow_{B'} t$ and $A' \equiv s \rightarrow_B t'$ follow similarly.

- For term reductions $t \rightsquigarrow t'$ we first split on the reduction. Note that as variables do not reduce, we must have that $t \equiv \mathsf{coh}\ (\Delta : A)[\sigma]$.

  – Argument reduction: If $t \rightsquigarrow t'$ is an argument reduction then $t' \equiv \mathsf{coh}\ (\Delta : A)[\sigma']$ where $\sigma \rightsquigarrow \sigma'$. By inductive hypothesis, $\mathsf{sd}(\sigma) > \mathsf{sd}(\sigma')$ and so $sd(t) > \mathsf{sd}(t')$.

  – Cell reduction: If $t \rightsquigarrow t'$ is a cell reduction then we have $t' \equiv \mathsf{coh}\ (\Delta : A')[\sigma]$ with $A \rightsquigarrow A'$. By inductive hypothesis $\mathsf{sd}(A) > \mathsf{sd}(A')$ and so $\mathsf{sd}(t) \geqslant \mathsf{sd}(t')$.

  – Insertion: If $t \rightsquigarrow t'$ is an insertion along a variable x then let $\sigma(x) \equiv \mathsf{coh}\ (\Theta : B)[\tau]$. Now let $\mu \equiv \sigma \ll x\ \tau$ and let $I \equiv \Delta \ll x\ \Theta$. Then:

$$\omega^{\dim(A)} \# \mathsf{sd}(\mu) = \omega^{\dim(A)} \# \#_{y \in \mathsf{Var}(I)} \mathsf{sd}(\mu(y))$$

$$= \omega^{\dim(A)} \# \#_{y \in \mathsf{Var}(I) \cap \mathsf{Var}(\Theta)} \mathsf{sd}(\mu(y)) \# \#_{y \in \mathsf{Var}(I) \cap \mathsf{Var}(\Delta)} \mathsf{sd}(\mu(y))$$

$$= \omega^{\dim(A)} \# \#_{y \in \mathsf{Var}(\Theta)} \mathsf{sd}(\tau(y)) \# \#_{y \in \mathsf{Var}(I) \cap \mathsf{Var}(\Delta)} \mathsf{sd}(\sigma(y))$$

$$= \omega^{\dim(A)} \# \mathsf{sd}(\tau) \# \#_{y \in \mathsf{Var}(I) \cap \mathsf{Var}(\Delta)} \mathsf{sd}(\sigma(y))$$

$$\geqslant \mathsf{sd}(\sigma(x)) \# \#_{y \in \mathsf{Var}(I) \cap \mathsf{Var}(\Delta)} \mathsf{sd}(\sigma(y))$$

$$= \#_{y \in (\mathsf{Var}(I) \cup \{x\}) \cap \mathsf{Var}(\Delta)} \mathsf{sd}(\sigma(x))$$

$$\leqslant \#_{y \in \mathsf{Var}(\Delta)} \mathsf{sd}(\sigma(y))$$

$$= \mathsf{sd}(\sigma)$$

Then by assumption, t is not a coherence on a disc, and so is not an identity. Therefore:

$$\mathsf{sd}(t) = \omega^{\dim(t)} \# \mathsf{sd}(A) \# \mathsf{sd}(\sigma)$$
$$\geqslant \omega^{\dim(t)} \# \mathsf{sd}(\sigma)$$
$$\geqslant \omega^{\dim(t)} \# \omega^{\dim(t)} \# \mathsf{sd}(\sigma \ll x\ \tau)$$
$$> \omega^{\dim(t)} \# \mathsf{sd}(A[\![\kappa]\!]) \# \mathsf{sd}((\sigma \ll x\ \tau))$$
$$= \mathsf{sd}(t')$$

where $\omega^{\dim(t)} > \mathsf{sd}(A[\![\kappa]\!])$ due to dimension.

This covers all cases, and so completes the proof. □

Theorem 44. *The reduction* $\rightsquigarrow$ *is strongly normalising on types, terms, and substitutions.*

Proof. Suppose an infinite reduction sequence $t_0 \rightsquigarrow t_1 \rightsquigarrow \ldots$ exists. Then by Lemma 43, $sd(t_0) > sd(t_1) > \ldots$ is an infinitely decreasing sequence of ordinals, which cannot exist. □

4.2 Confluence

In this section we show that $\rightsquigarrow'$ is locally confluent, which implies along with Theorem 44 that $\rightsquigarrow'$ is globally confluent and hence complete.

Definition 45. Mutually define a term t being *regular* and its *regular height*, an extended natural number as follows:

- If t is a variable then it is regular and its regular height is ∞

- If $t \equiv \mathsf{coh}\ (\Delta : U)[\sigma]$, then t is regular if and only if Δ is not a disk, U is the unbiased type over Δ, every term in σ is regular, and for all $x \in \mathsf{LM}(\Delta)$, the branching height of x is less than the regular height of $\sigma(x)$. If t is regular, then its regular height is given by the linear height of Δ.

If I, ι_i, ι_e is an insertion of (Θ, B) into (Δ, x), with U is the unbiased type over Δ and B is the unbiased type over Θ, then all terms contained in $U[\![\iota_e]\!]$ are regular, as all the terms in U are unbiased composites of variables, and we substitute in unbiased composites which satisfy the height condition from the conditions of insertion.

Further note that if the support of a regular term is a disc, then it must be a variable, as any unbiased composite will have multiple locally maximal variables in its support.

Lemma 46. *If t is a regular term over a pasting diagram, then it reduces to the unbiased composite of its support.*

Proof. Proceed by mutual reduction on dimension and subterms. The hypothesis on dimension tells us that if a regular term undergoes an insertion, then it reduces to another regular term, (in other words the type reduces to the unbiased type). Further, we must have that the regular height of the new reduced term is at least the regular height of the original term. Therefore take an arbitrary term t, and repeatedly perform insertions followed by making everything unbiased. By termination, we will eventually get term t' with $t \rightsquigarrow t'$ and t' having no more possible insertions.

If t' is a variable then we are done. Otherwise, t' is an unbiased composition. By construction, t' does not admit any insertions, so all locally maximal dimensional arguments must be variables (or t' would not be regular). Now suppose there is an argument x is not a variable. Then there is such an argument u with maximal dimension. This can not be locally maximal so it is the source or target of a variable. Then its support must be a disc, but this contradicts its regularity. □

Corollary 47. *If t is an unbiased composite and admits an insertion, then the resulting term reduces to an unbiased composite.*

Theorem 48. *The reduction $\rightsquigarrow'$ is locally confluent. In other words if a, b, and c are valid terms, with $a \rightsquigarrow b$ and $a \rightsquigarrow c$ then there exists valid u with $b \rightsquigarrow_{\mathsf{rt}} u$ and $c \rightsquigarrow_{\mathsf{rt}} u$.*

Proof. Proceed by mutual induction on subterms and dimension. By induction on dimension we have that $\rightsquigarrow$ is locally confluent on m-dimensional terms for $m < \dim(a)$ and so is globally confluent (for dimension m). Further, as $\rightsquigarrow$ is terminating (by Theorem 44) and agrees with equality (by Proposition 39), we have that any equal terms with dimension less than $\dim(a)$ have a common normal form. It also holds that if $s =_m t$ for some $m < \dim(a)$, then there is u with $s \rightsquigarrow_{\mathsf{rt}} u$ and $t \rightsquigarrow_{\mathsf{rt}} u$.

Suppose $a \rightsquigarrow b$ and $a \rightsquigarrow c$. Split into cases.

- Both are insertions. If they are insertions on the same variable, then $b \equiv c$ and we are done. Otherwise let $a \equiv \mathsf{coh}\,(\Gamma : A)[\sigma]$, and suppose b is an insertion along x with $\sigma(x) \equiv \mathsf{coh}\,(\Delta : B)[\tau]$ and c is an insertion along y with $\sigma(y) \equiv \mathsf{coh}\,(\Theta : C)[\mu]$. Further we must have that x and y are locally maximal, so in particular are not in each other's support.

$$
\begin{array}{ccc}
D^n \xrightarrow{\overline{\mathsf{coh}\,(\Delta:B)[\mathrm{id}]}} \Delta & & \\
\big\downarrow{\overline{x}} & {}^{\llcorner} & \big\downarrow \\
D^m \xrightarrow{\ \overline{y}\ } \Gamma & \longrightarrow & \Gamma \ll x\ \Delta \\
{\scriptstyle\overline{\mathsf{coh}\,(\Theta:C)[\mathrm{id}]}}\big\downarrow & & \\
\Theta & &
\end{array}
$$

In the diagram above we construct the insertion along x. Now we can notice that the composite $D^m \xrightarrow{y} \Gamma \to \Gamma \ll x\ \Delta$ is just the substitution $\overline{y}$, and so we can perform another insertion (as the branching height of y must be the same in Γ and $\Gamma \ll x\ \Delta$). We then get the following diagram:

$$
\begin{array}{ccc}
D^n \xrightarrow{\overline{\mathsf{coh}\,(\Delta:B)[\mathrm{id}]}} \Delta & & \\
\big\downarrow{\overline{x}} & {}^{\llcorner} & \big\downarrow \\
D^m \xrightarrow{\ \overline{y}\ } \Gamma & \longrightarrow & \Gamma \ll x\ \Delta \\
{\scriptstyle\overline{\mathsf{coh}\,(\Theta:C)[\mathrm{id}]}}\big\downarrow & {}^{\llcorner} & \big\downarrow \\
\Theta & \longrightarrow & (\Gamma \ll x\ \Delta) \ll y\ \Theta
\end{array}
$$

and therefore $(\Gamma \ll x\ \Delta) \ll y\ \Theta$ is the colimit of:

$$
\begin{array}{cc}
& D^n \xrightarrow{\overline{\mathsf{coh}\,(\Delta:B)[\mathrm{id}]}} \Delta \\
& \big\downarrow{\overline{x}} \\
D^m \xrightarrow{\ \overline{y}\ } \Gamma & \\
{\scriptstyle\overline{\mathsf{coh}\,(\Theta:C)[\mathrm{id}]}}\big\downarrow & \\
\Theta &
\end{array}
$$

Similarly, $(\Gamma \ll y\ \Theta) \ll x\ \Delta$ is a colimit of the same diagram and so is equal. Further, it can easily be checked that $(\sigma \ll y\ \mu) \ll x\ \tau$ and $(\sigma \ll x\ \tau) \ll y\ \mu$ both satisfy the necessary equations to be equal by uniqueness of the map out of a colimit. It is then simple to check that the

maximal dimension parts of these two substitutions are actually definitionally equal by case analysis and so we get

$$b \rightsquigarrow \mathsf{coh} \; ((\Gamma \ll x \; \Delta) \ll y \; \Theta : A[\![\kappa_{(\Gamma \ll x \; \Delta),y,\Theta} \circ \kappa_{\Delta,x,\Gamma}]\!])[(\sigma \ll x \; \tau) \ll y \; \mu]$$

$$=_{\dim(a)} \mathsf{coh} \; ((\Gamma \ll y \; \Theta) \ll x \; \Delta : A[\![\kappa_{(\Gamma \ll y \; \Theta),x,\Delta} \circ \kappa_{\Theta,y,\Gamma}]\!])[(\sigma \ll y \; \mu) \ll x \; \tau]$$

$$\twoheadleftarrow c$$

which gives us a way to construct a suitable u by the comment at the start of the proof.

- Now suppose both reductions are argument reductions. If these occur on different arguments, we simply apply both to get a suitable u, and if they are the same argument then we use the inductive hypothesis on subterms.

- Now suppose $a \rightsquigarrow b$ is an insertion and $a \rightsquigarrow c$ is an argument reduction. The only case which is hard is when the insertion and argument reduction are along the same variable. Therefore we can assume we are in the case where $a \equiv \mathsf{coh} \; (\Gamma : A)[\sigma]$ and $\sigma(x) \equiv \mathsf{coh} \; (\Delta : B)[\tau]$ and $\tau(y) \equiv \mathsf{coh} \; (\Theta : C)[\mu]$.

Firstly assume that Δ is a disc context. Then $\sigma(x)$ just reduces to $\tau(y)$ by performing the argument insertion first. If instead the head insertion is performed first then the head pasting diagram is unchanged (up to variable renaming) but the arguments can be changed. We see that x must be sent to $\tau(y)$. Therefore $b =_{\dim(x)} c$ and so we can find suitable u as before.

Now we assume Δ is not a disc context. First consider the case where the insertion is done first. Then we get the following diagram:

$$
\begin{array}{ccc}
D^n & \xrightarrow{\;\overline{x}\;} & \Gamma \\
{\scriptsize \overline{\mathsf{coh} \; (\Delta:B)[\mathrm{id}]}} \downarrow & \quad \ulcorner & \downarrow \\
D^m \xrightarrow{\;\overline{y}\;} \Delta & \longrightarrow & \Gamma \ll x \; \Delta \\
{\scriptsize \overline{\mathsf{coh} \; (\Theta:C)[\mathrm{id}]}} \downarrow & \quad \ulcorner & \downarrow \\
\Theta & \longrightarrow & (\Gamma \ll x \; \Delta) \ll y \; \Theta
\end{array}
$$

where we can perform the second insertion as the composite $D^m \xrightarrow{y} \Delta \to \Gamma \ll x \; \Delta$ is just the variable y (as $\Delta \to \Gamma \ll x \; \Delta$ is variable to variable) and y has the same branching height in Δ and $\Gamma \ll x \; \Delta$ as Δ is not a disc context.

Now consider first applying the argument insertion to get this diagram:

$$
\begin{array}{ccc}
& D^n \xrightarrow{\;\overline{x}\;} \Gamma \\
& {\scriptsize \overline{\mathsf{coh} \; (\Delta:B)[\mathrm{id}]}} \downarrow \\
D^m \xrightarrow{\;\overline{y}\;} \Delta \\
{\scriptsize \overline{\mathsf{coh} \; (\Theta:C)[\mathrm{id}]}} \downarrow & \quad \ulcorner & \downarrow \\
\Theta \longrightarrow \Delta \ll y \; \Theta
\end{array}
$$

23

Now note that there is a second map from D^n to $\Delta \lll y\ \Theta$ given by $\mathsf{coh}\ (\Delta \lll y\ \Theta : U)[\mathrm{id}]$ where U is the unbiased type of $\Delta \lll y\ \Theta$ and by an insertion and Corollary 47 these are equal. As Δ is not a disc context, the linear height of $\Delta \lll y\ \Theta$ is at least as high as Δ and so we can complete the second insertion to get:

$$
\begin{array}{ccccc}
 & & D^n & \xrightarrow{\;\;\overline{x}\;\;} & \Gamma \\
 & {\scriptstyle \overline{\mathsf{coh}\ (\Delta:B)[\mathrm{id}]}}\downarrow & & & \downarrow \\
D^m & \xrightarrow{\;\overline{y}\;} & \Delta & = \quad \overline{\mathsf{coh}\ (\Delta \lll y\ \Theta : U)[\mathrm{id}]} & \\
{\scriptstyle \overline{\mathsf{coh}\ (\Theta:C)[\mathrm{id}]}}\downarrow & & \downarrow & & \downarrow \\
\Theta & \longrightarrow & \Delta \lll y\ \Theta & \longrightarrow & \Gamma \lll x\ (\Delta \lll y\ \Theta)
\end{array}
$$

Then as before, both of these insertions arise from the same colimit and we can apply the same reasoning as the insertion-insertion case, which completes the proof.

$\square$

5 Typechecking

Using the reduction in the previous section, we can show that equality and typechecking are decidable.

Definition 49. Define the normalisation function N as a function taking terms to terms, types to types, and substitutions to substitutions, given by repeatedly applying the innermost leftmost reduction until a normal form is found.

Proposition 50. *If s and t are valid terms, then $s = t$ if and only if $N(s) \equiv N(t)$. A similar result holds for valid typed A and B, and valid substitutions σ and τ.*

This gives us a way to decide equality between terms which are valid, as we can simply compute the normalisation of each term and checking syntactic equality is clearly decidable. We can now show how this can be used to perform decidable typechecking.

Theorem 51. *Typechecking on terms, types, substitutions, and contexts is decidable.*

Proof. We work by mutual induction on terms, types, substitutions, and contexts.

- For typechecking whether $\Gamma \vdash t : A$, we split into cases:

  - If t is a variable, then we check whether $t \in \Gamma$. If it is not, then t is not valid in Γ. If $t : B$ in Γ then we just need to check if $A = B$.

  - If $t \equiv \mathsf{coh}\ (\Delta : B)[\sigma]$, then for t to be valid, it is necessary that $\Gamma \vdash \sigma : \Delta$, $\Delta \vdash_p$, and $\Delta \vdash B$. It can be easily checked that Δ is a pasting diagram, and the other two can be checked by inductive hypothesis. We then need to further check that the support conditions hold for either the coherence or composition rule. Finally, if all other conditions hold then $\Gamma \vdash t : A$ if and only if $A = B[\![\sigma]\!]$.

- For typechecking $\Gamma \vdash A$, if $A \equiv \star$ then A is valid, and if $A \equiv s \to_B t$, then A is valid if and only if $\Gamma \vdash B$, $\Gamma \vdash s : B$, and $\Gamma \vdash t : B$, all of which can be checked by inductive hypothesis.

24

- Typechecking for substitutions similarly follows from inductive hypothesis.

- The empty context is always valid, and $\Gamma, x : A$ can be checked by checking whether $\Gamma \vdash$ and $\Gamma \vdash A$.

This gives us a fully decidable procedure for typechecking. It can also be noted that we can extract a type inference algorithm for terms from this proof. $\qquad\square$

References

[1] Thibaut Benjamin.
    "A type theoretic approach to weak ω-categories and related higher structures".
    Theses. Institut Polytechnique de Paris, Nov. 2020.
    URL: https://tel.archives-ouvertes.fr/tel-03106197.

[2] Thibaut Benjamin, Eric Finster, and Samuel Mimram.
    "Globular weak ω-categories as models of a type theory". 2020. URL:
    http://www.lix.polytechnique.fr/~tbenjamin/articles/publications/models.pdf.

[3] Christoph Dorn. "Associative n-categories". In: *CoRR* (2018).
    URL: http://arxiv.org/abs/1812.10586v1.

[4] Eric Finster and Samuel Mimram.
    "A Type-Theoretical Definition of Weak Omega-categories". In: *CoRR* (2017).
    URL: http://arxiv.org/abs/1706.02866v1.

[5] Eric Finster, David Reutter, and Jamie Vicary.
    "A Type Theory for Strictly Unital ∞-Categories". In: *CoRR* (2020).
    URL: http://arxiv.org/abs/2007.08307v1.

[6] R. Gordon, A. J. Power, and Ross Street. *Coherence for tricategories*. English.
    558th ed. Vol. 117. United States: American Mathematical Society, Sept. 1995.

[7] Lukas Heidemann, Nick Hu, and Jamie Vicary. homotopy.io. 2019.
    URL: https://doi.org/10.5281/zenodo.2540764.

[8] André Joyal and Joachim Kock. *Weak Units and Homotopy 3-types*. 2006.

[9] Joachim Kock.
    "Note on Commutativity in Double Semigroups and Two-Fold Monoidal Categories".
    In: *CoRR* (2006). URL: http://arxiv.org/abs/math/0608452v1.

[10] Tom Leinster. *A Survey of Definitions of n-Category*. 2001.

[11] Paolo Lipparini. "An Infinite Natural Sum".
    In: *Mathematical Logic Quarterly* 62.3 (2016), pp. 249–257.
    URL: https://doi.org/10.1002/malq.201500017.

[12] Georges Maltsiniotis.
    "Grothendieck ∞-groupoids, and still another definition of ∞-categories".
    In: *arXiv preprint arXiv:1009.2331* (2010).

[13] David Reutter and Jamie Vicary.
    "High-level methods for homotopy construction in associative n-categories".
    In: *arXiv preprint arXiv:1902.03831* (2019).

[14]   Carlos Simpson. "Homotopy Types of Strict 3-groupoids". In: *CoRR* (1998).
       URL: http://arxiv.org/abs/math/9810059v1.