UNIVERSITY OF BIRMINGHAM

High-level methods for homotopy construction in associative n-categories

Reutter, David; Vicary, Jamie

Link to publication on Research at Birmingham portal

High-level methods for homotopy construction in associative n-categories

David Reutter [1]

david.reutter@cs.ox.ac.uk

Jamie Vicary [1][2]

vicaryjo@cs.bham.ac.uk

Abstract—A combinatorial theory of associative n-categories has recently been proposed, with strictly associative and unital composition in all dimensions, and the weak structure arising as a notion of 'homotopy' with a natural geometrical interpretation. Such a theory has the potential to serve as an attractive foundation for a computer proof assistant for higher category theory, since it allows composites to be uniquely described, and relieves proofs from the bureaucracy of associators, unitors and their coherence. However, this basic theory lacks a high-level way to construct homotopies, which would be intractable to build directly in complex situations; it is not therefore immediately amenable to implementation.

We tackle this problem by describing a 'contraction' operation, which algorithmically constructs complex homotopies that reduce the lengths of composite terms. This contraction procedure allows building of nontrivial proofs by repeatedly contracting subterms, and also allows the contraction of those proofs themselves, yielding in some cases single-step witnesses for complex homotopies. We prove correctness of this procedure by showing that it lifts connected colimits from a base category to a category of zigzags, a procedure which is then iterated to yield a contraction mechanism in any dimension. We also present *homotopy.io*, an online proof assistant that implements the theory of associative n-categories, and use it to construct a range of examples that illustrate this new contraction mechanism.

I. INTRODUCTION

The theory of associative n-categories (ANCs) has recently been proposed [1, 2]. As with the theory of strict n-categories [3], composition in this theory is strictly associative and unital in all dimensions. However, unlike the strict theory, ANCs retain a significant amount of weak structure—in the form of homotopies[3], with a natural geometric interpretation—making it reasonable to conjecture that every weak n-category is equivalent to an ANC [1, Conjecture I.5.0.4].[4]

It may therefore be possible for ANCs to serve as an attractive general language for calculations in higher category theory, if suitably encoded into a computer proof assistant. Strict associators and units would make composites unique, eliminating some of the bureaucracy of coherence, while the remaining weak structure—while still potentially of high complexity—could be reasoned about geometrically.

The major obstacle to realizing this goal is the difficulty of constructing nontrivial terms of the theory. In principle these can encode complex data, including not only the composites of generating types in arbitrary dimension, but also arbitrary homotopies of these composites. Each term has a dimension, and the n-dimensional terms are called n-*diagrams*.

As examples of such terms, consider Figures 1 and 2. Figure 1 shows two 2-diagrams, which can be interpreted as expressions in the string diagram calculus for a monoidal category [5]; the arrow represents a homotopy of these 2-diagrams, and forms part of the data of a 3-diagram. Figure 2 shows two 3-diagrams, which can be interpreted as tangles [6] in the string diagram calculus for a braided monoidal category; the arrow represents a homotopy of these 3-diagrams, and forms part of the data of a 4-diagram.

---

[3]*Homotopy* is a standard notion from algebraic topology, which can be understood informally to mean the continuous deformation of one topological structure into another. We use the term only in an informal sense, basing our formal mathematical development on the theory of associative n-categories, which have a combinatorial foundation.

[4]In dimension 3 the theory of ANCs agrees with the theory of Gray categories, a well-known model of 3-categories which is equivalent to the fully-weak theory [4], but which has strict associators and unitors; for $n > 3$, ANCs are not expected to be equivalent to any previously-described theory.

---

[1]Department of Computer Science, University of Oxford, UK
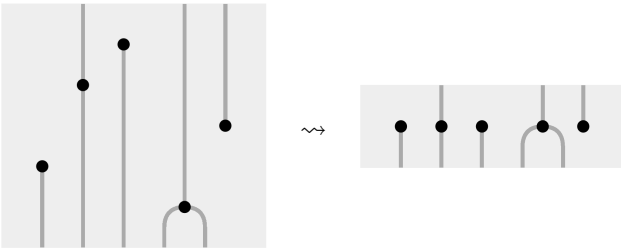[2]School of Computer Science, University of Birmingham, UK

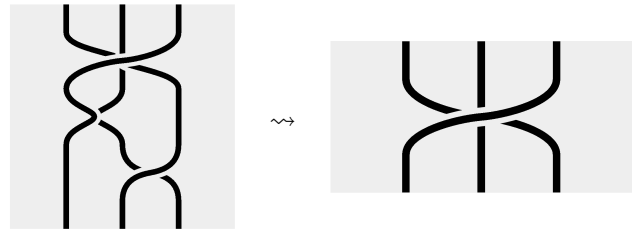Fig. 1: A 3-dimensional homotopy of 2-dimensional diagrams. *(Link to online proof)*

Fig. 2: A 4-dimensional homotopy of 3-dimensional diagrams. *(Link to online proof)*

The main mathematical contribution of this paper is the description of a *contraction* algorithm, which builds homotopies that reduce a given portion of the diagram in size, along with a full mathematical theory that demonstrates correctness of the procedure. This gives a high-level method for building nontrivial homotopies in an associative n-category. Figures 1 and 2 both give examples; in each case, the second diagram was obtained by executing the contraction procedure on the first diagram.

Contraction can serve as the main workhorse for the construction of a range of nontrivial proofs in the theory. Given an initial composite n-diagram, we produce our $(n + 1)$-dimensional proof object by contracting various k-dimensional subdiagrams for $k \leqslant n$ to produce the content of the $(n+1)$-dimensional proof object, as well as applying algebraic moves from the signature, and extending these recursively to the diagram as a whole using some further techniques. Once our proof is complete, we can then contract that proof term itself, to yield a shorter proof of the same logical statement. In Section IV we give two fully-worked examples of this entire proof construction workflow. Indeed, we conjecture that contractions, together with the associated recursive techniques, yield a universal toolkit which can in principle construct any homotopy in the theory.

We also present *homotopy.io* [7], an online proof assistant that implements the theory of associative n-categories. This proof assistant is enabled by our new theory of contractions, which serves as the main engine for homotopy construction, and is applied by clicking and dragging with the mouse on the graphical representation rendered by the tool. We present the tool as an accompaniment to the claims of the paper, demonstrating that the theory of contractions that we build here is useful and practical.

We keep the focus of this paper on logical foundations, and do not give further details on the implementation. Nonetheless, we accompany many of our examples with direct hyperlinks to their online formalization in the tool, which we invite the reader to investigate. To explore these workspaces, change the parameters of the "Slice" control at the top-right of the window; to manipulate the diagrams homotopically, use the mouse to drag vertices (or crossings) up or down, or drag wires left or right.

A. Related work

This work builds on the existing theory of ANCs due to Dorn, Douglas and Vicary [1, 2]. That theory defines *signatures* that give families of admissible types, *diagrams* that encode composites and homotopies of these types, *term normalization* which reduces a diagram to a standard form, and *type checking* which verifies whether a normalized diagram is valid. The tool *homotopy.io* implements all these core aspects of the theory, about which we give no further details in this paper. However, that existing theory does not include the concept of contraction, or yield any other high-level method for homotopy construction, motivating our results here.

The theory of ANCs can be seen as a development into arbitrary dimension of the theory of quasistrict 4-categories of Bar and Vicary [8], implemented as the proof assistant *Globular* by Bar, Kissinger and Vicary [9]. That proof assistant had a restricted notion of homotopy construction, limited fundamentally to dimension 4, and could not even in principle be generalized to arbitrary dimension, where our results apply.

Having in hand a high-level method for homotopy construction in arbitrary dimension, it is interesting to ask for an algorithm which, given a pair of n-diagrams, either constructs a correct homotopy between them, or correctly reports that no such homotopy exists. Such an algorithm was recently given for the case of 2-diagrams [10], running in quadratic time. The general case is known to be decidable by work of Makkai [11].

Contractions, as we present them in this paper, are colimit constructions for sequences of cospans. Spans and cospans have seen wide application in the theory of higher categories, in particular by Baez and collaborators [12], Grandis [13], Morton [14] and Stay [15]; however, in these approaches, a colimit construction usually yields cospan *composition* is often given as a colimit construction. This highlights a key difference: in our work, we compose cospans just by arranging them side-by-side, with the colimits—which do not always exist—instead giving us the high-level contraction structure.

B. Overview of the paper

Our contribution is structured as follows. In Section II we introduce zigzag categories and explore their properties, culminating in simple definitions of untyped and typed n-diagrams. In Section III we give a construction procedure for colimits on a zigzag category in terms of colimits in the base category, prove its correctness, and show how this gives rise to a contraction procedure for diagrams. In Section IV, we show how contraction works together with some other simple mechanisms to give a general toolkit for homotopy construction, and we give a wide range of examples.

C. Notation

For a natural number $n \in \mathbb{N}$, we write $[n]$ for the totally-ordered set $\{0, 1, \ldots, n - 1\}$. We use boldface capital letters $\mathbf{A}, \mathbf{B}, \mathbf{C}, \ldots$ for categories. We write $\mathbf{1}$ for the terminal category, with unique object $\bullet$ and only the identity morphism, and Δ for the category of (possibly empty) finite totally-ordered sets and order-preserving functions.

D. Acknowledgements

II. ZIGZAG CATEGORIES

Our theory is based on the notion of zigzag, a reworking and simplification of the notion of *singular interval* from [1, 2], and zigzag maps, corresponding to the notion of *limit* in that reference. In this section we develop the theory of zigzags and their maps, and show how they can be used to give definitions of untyped and typed n-dimensional diagrams.
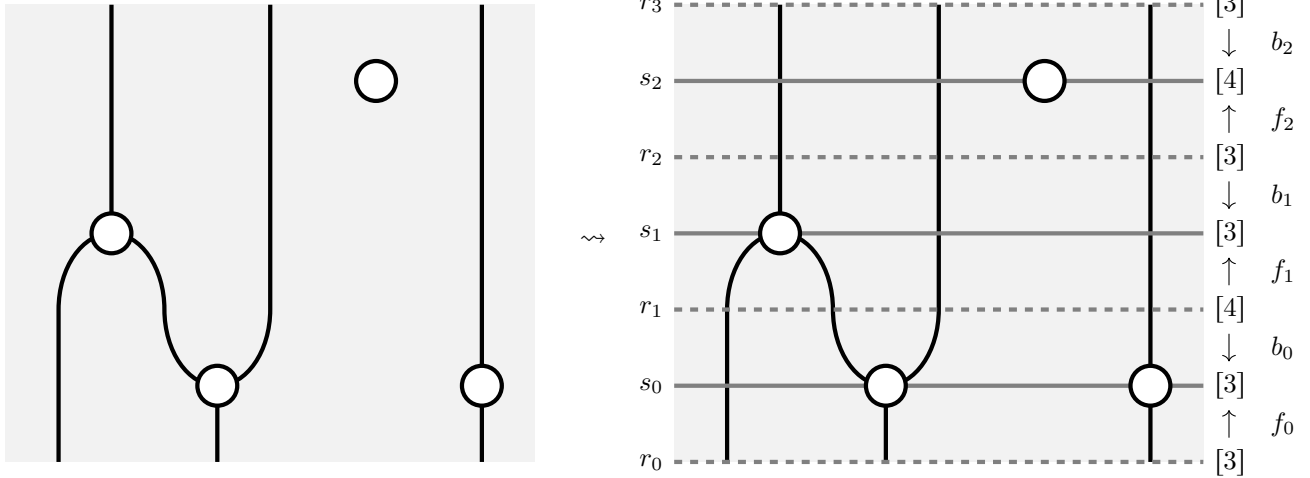
Fig. 3: Decomposing a 2-dimensional string diagram as a sequence of monotone functions.

A. Motivation

We motivate the theory of zigzags by examining a 2-dimensional string diagram, as illustrated on the left of Figure 3, drawn in the standard Joyal-Street graphical calculus for monoidal categories [5, 16], and considering how we could represent it combinatorially. At 3 distinct heights, the diagram contains vertices (which we imagine to be pointlike); we call these the *singular heights*, and label them s_0, s_1 and s_2.

If we formally remove these heights, we disconnect the diagram into 4 sections, none of which contain any vertices. For any such section, the geometrical content of any two heights will be equivalent; in particular, the number of wires present at two such heights must be the same, since wires are only created or destroyed by vertices. So we arbitrarily choose one height in each of these sections, called the *regular heights*, and label them r_0, r_1, r_2 and r_3.

We now have 7 chosen heights, and at each of them we count how many geometrical entities (either vertices or wires) are present at that height. For example, r_0 intersects 3 entities (all wires), and s_2 intersects 4 entities (3 wires and 1 vertex). These entities form a totally-ordered set in a natural way, in their order of appearance from left-to-right within each height, and we write the corresponding totally-ordered set $[n]$ at the right of the diagram.

We then choose a regular height r_i, and imagine it converging to one of its adjacent singular heights s_j. This process will induce an order-preserving function from the entities at height r_i to the entities at height s_j, and we write these functions as f_0, b_0, f_1, b_1, etc, to the side of the diagram.

From our original diagram, we have therefore obtained a family of totally-ordered sets, and monotone functions between them, with an alternating pattern of directions. This is an instance of the general theory of *singular intervals* [1, 2], and directly motivates the more elementary theory of *zigzags*, which we now explore.

B. Zigzags and zigzag maps

Definition 1. In a category $\mathbf{C}$, a *zigzag* Z is a finite diagram of the following sort:

$$
\begin{array}{ccccccccc}
& s_0 & & s_1 & & & & s_{n-1} & \\
f_0 \nearrow & & \nwarrow b_0 \; f_1 \nearrow & & \nwarrow b_1 & \ldots & f_{n-1} \nearrow & & \nwarrow b_{n-1} \\
r_0 & & r_1 & & & & & & r_n
\end{array} \quad (1)
$$

We write $Z_{\mathrm{sing}} = [n]$ for the ordered set of *singular heights*, and $Z_{\mathrm{reg}} = [n+1]$ for the ordered set of *regular heights*. The objects $r_0, r_1, \ldots$ are called the *regular objects*, and the objects $s_0, s_1, \ldots$ are called the *singular objects*. Such a zigzag has *length* n, given by the number of singular heights. Zigzags of length 0 are allowed, and consist of a single regular height only, and no morphisms. We write $f_i : r_i \to s_i$ and $b_i : r_{i+1} \to s_i$ for the *forward and backward morphisms* in the diagram as indicated, for all $i \in Z_{\mathrm{sing}}$. Where it might be unclear which zigzag Z we are referring to, we will write $r_i^Z, s_i^Z, f_i^Z, b_i^Z$ instead of r_i, s_i, f_i, b_i.

Before we can define maps of zigzags, we need a short formal development.

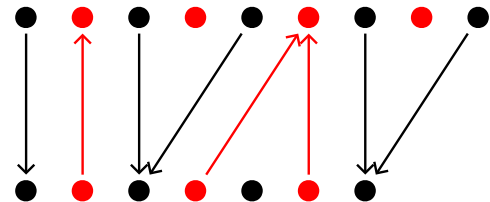**Definition 2.** Let $(-)^{\mathrm{T}} : \Delta \to \Delta$ be the functor that adds

Fig. 4: An interleaved illustration of a monotone map $f : [3] \to [4]$ in Δ going up the page in red, and $f' : [5] \to [4]$ in $\Delta_=$ going down the page in black.
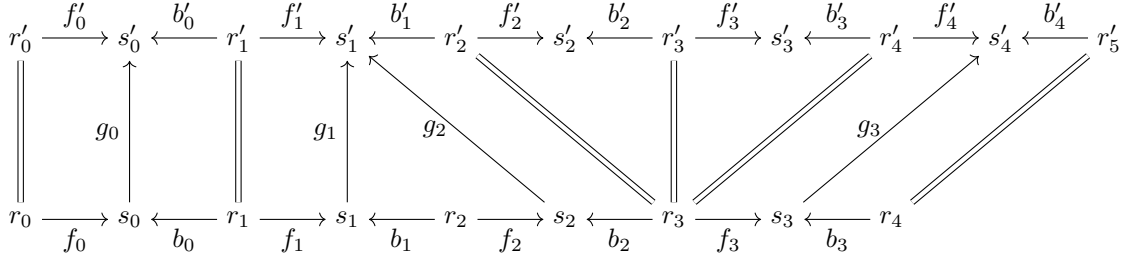
3

Fig. 5: A zigzag map diagram $g : Z \to Z'$, running bottom-to-top.

an element "on top" of the total orders, acting on objects as $n \mapsto n + 1$, and on a monotone map $f : [n] \to [m]$ as $f^{\mathrm{T}}(n) = m$ and $f^{\mathrm{T}}(i) = f(i)$ for $0 \leqslant i < n$.

Definition 3. Given a monotone map $f : [n] \to [m]$, for any element $j \in [m]$ in the target, define $f_{\geqslant j} = \{i \in [n] | f(i) \geqslant j\}$ as the elements in the source whose image is above j.

For any monotone map $f : [n] \to [m]$, for any $j \in [m]$, note that that $(f^{\mathrm{T}})_{\geqslant j}$ is always nonempty, due to the additional top element, and thus $\max\left((f^{\mathrm{T}})_{\geqslant j}\right)$ is well-defined.

Definition 4. The category $\Delta_=$ has nonzero natural numbers as objects, and as morphisms $n \to m$, monotone maps $[n] \to [m]$ preserving the first and last elements.

Definition 5. The functor $(-)' : \Delta \to \Delta_=^{\mathrm{op}}$ acts on objects as $n \mapsto n + 1$, and acts on morphisms $f : [n] \to [m]$ as $f'(j) = \max\left((f^{\mathrm{T}})_{\geqslant j}\right)$.

We illustrate this in Figure 4, which shows a monotone map f in red, and its "reversal" f' in black.

Lemma 6. *The functor $(-)'$ is an equivalence.*

Proof. The functor is clearly surjective on objects. That it is fully faithful can be seen by inspection of Figure 4: from any black right-to-left monotone preserving first and last elements, a red left-to-right monotone can be constructed by "filling in the gaps", and vice-versa. □

Abusing notation, we also denote the inverse of this functor $\Delta_= \to \Delta$ by $(-)'$, restricting its use in this way to situations where there is no ambiguity.

We now use this technology to define maps of zigzags.

Definition 7. In a category $\mathbf{C}$, a *zigzag map* $f : Z \to Z'$ comprises a monotone function $f_{\mathrm{sing}} : Z_{\mathrm{sing}} \to Z'_{\mathrm{sing}}$, and for each $i \in Z_{\mathrm{sing}}$ a morphism $f_i : s_i \to s'_{f_{\mathrm{sing}}(i)}$. Defining $f_{\mathrm{reg}} = (f_{\mathrm{sing}})' : Z'_{\mathrm{reg}} \to Z_{\mathrm{reg}}$, we then require that the diagram constructed as follows, which can always be laid out in a planar way, is commutative:

(1) Take the disjoint union of Z and Z' as diagrams in $\mathbf{C}$.

(2) For every $i \in Z_{\mathrm{sing}}$, add the arrow f_i to the diagram, going from $s \in Z_{\mathrm{sing}}$ to $f_{\mathrm{sing}}(s) \in Z'_{\mathrm{sing}}$.
(3) For every $j \in Z'_{\mathrm{reg}}$, add an identity arrow to the diagram, between $j \in Z'_{\mathrm{reg}}$ and $f_{\mathrm{reg}}(j) \in Z_{\mathrm{reg}}$.

This construction is quite simple to use in practice. We illustrate it in Figure 5. Informally, it amounts to the following. (1) Draw the zigzags Z and Z' one above the other. (2) For each singular object of Z, add an arrow to some singular object of Z', such that the implied function $Z_{\mathrm{sing}} \to Z'_{\mathrm{sing}}$ is monotone. (3) In the spaces between these arrows, add all possible equalities between regular objects of Z and Z'.

In the example of Figure 5, Z has length 4 and Z' has length 5, with $g : Z \to Z'$ running bottom-to-top. The monotone $g_{\mathrm{sing}} : [4] \to [5]$ acts as $0 \mapsto 0$, $1 \mapsto 1$, $2 \mapsto 1$ and $3 \mapsto 4$, with the morphisms g_0, g_1, g_2 and g_3 having source and target objects as indicated. The equalities between regular heights force equalities of regular objects $r_0 = r'_0$, $r_1 = r'_1$, $r_3 = r'_2 = r'_3 = r'_4$, and $r_4 = r'_5$. The diagram is formed from 9 squares, all of which must commute, leading in this case to the requirements $f'_0 = g_0 \circ f_0$, $f'_0 = g_0 \circ b_0$, $f'_1 = g_1 \circ f_1$, $f_1 \circ b_1 = g_2 \circ f_2$, $b'_1 = g_2 \circ b_2$, $f'_2 = b'_2$, $f'_3 = b'_3$, $b'_4 = g_3 \circ f_3$, and $f'_4 = g_3 \circ b_3$.

Zigzags and their maps form a category in the obvious way.

Definition 8. Given a category $\mathbf{C}$, the *zigzag category* $Z_{\mathbf{C}}$ is defined to have zigzags as objects and zigzag maps as morphisms.

Composition, associativity and units are clear. We will often be interested in iterating this construction, as follows.

Definition 9. Given a category $\mathbf{C}$, the *iterated zigzag category* $Z_{\mathbf{C}}^n$ is the category obtained by starting with the category $\mathbf{C}$, and taking the zigzag category n times.

Every zigzag category has forgetful functors to Δ and $\Delta_=^{\mathrm{op}}$.

Definition 10 (Regular and singular monotone functors). For a category $\mathbf{C}$, the *singular monotone functor* $S_{\mathbf{C}} : Z_{\mathbf{C}} \to \Delta$ acts as $S_{\mathbf{C}}(Z) = Z_{\mathrm{sing}}$ and $S_{\mathbf{C}}(f) = f_{\mathrm{sing}}$, and the *regular monotone functor* $R_{\mathbf{C}} : Z_{\mathbf{C}} \to \Delta_=^{\mathrm{op}}$ acts as $R_{\mathbf{C}} = (-)' \circ S_{\mathbf{C}}$.

Example 11. Starting with the terminal category $\mathbf{1}$, we see that $Z_{\mathbf{1}} = \Delta$, the singular monotone functor $S_{\mathbf{1}} : \Delta \to \Delta$ is the identity, and the regular monotone functor $R_{\mathbf{1}} = (-)'$.

C. Untyped and typed diagrams

We can use zigzag categories to give a straightforward notion of untyped n-diagram, yielding an elementary untyped version of the diagrams which form the terms of the theory of associative n-categories [1].

Definition 12. An *untyped n-diagram* is an object of the iterated zigzag category $Z_{\mathbf{1}}^n$.

We explore this definition through the following examples, illustrated in Figure 6.

Example 13. The only untyped 0-diagram is $\bullet$, the point.

Example 14. An untyped 1-diagram is an object of $Z_{\mathbf{1}} = \Delta$, a finite ordinal. So the only parameter is the length of the composite.

Example 15. An untyped 2-diagram is an object of $Z_{Z_{\mathbf{1}}} = Z_{\Delta}$.

Note in particular that the untyped 2-diagram illustrated in Figure 6(c) corresponds exactly to the example of Figure 3, which motivated our construction in the first place.

To develop the theory of typed diagrams, suppose that L is a set of labels equipped with an arbitrary *dimension function* $\mathrm{d} : L \to \mathbb{N}$. The intuition is that L is a set of generators for a higher category, and the dimension function assigns to each generator its dimension as a cell. We can then build a poset $\mathbf{L}$, whose objects are elements of the set L, and where for any $l, l' \in L$, there is a morphism $l \to l'$ in $\mathbf{L}$ just when $l = l'$ or $\dim(l) < \dim(l')$. We can use this to give a generalization of Definition 12 appropriate for the typed setting.

Definition 16. For a set of types L equipped with a dimension function $\mathrm{d} : L \to \mathbb{N}$, an *$L$-typed n-diagram* is an object of the iterated zigzag category $Z_{\mathbf{L}}^n$.

An L-typed n-diagram is a similar structure to an untyped n-diagram, except that every "bottom-level point" is assigned an element of the label set L, in a way which is arbitrary, except that as we pass from one type to another along a zigzag map, the types must either stay the same, or increase in dimension. Indeed, it is clear that if we choose $L = \{\bullet\}$ with dimension function $\bullet \mapsto 1$, we we recover the theory of untyped n-diagrams as a special case.

The full theory of associative n-categories [1, 2] has a notion of *type signature* Σ, which defines a set of *type labels* $|\Sigma|$, and for each label $l \in |\Sigma|$ a *canonical neighbourhood*. There is then a *type checking scheme*, which takes as input a $|\Sigma|$-typed n-diagram, and returns a boolean, indicating whether or not it is well-typed with respect to Σ; that is, whether for every instance of every type label in the diagram, its neighbourhood in the diagram normalizes to its canonical neighbourhood. We do not discuss this further, as our contraction procedure operates just at the level of the categories $Z_{\mathbf{L}}^n$.

In the remainder of the paper, the n-diagrams we will draw will be typed, and therefore objects of $Z_{\mathbf{L}}^n$ for some label set L, about which we will not give details. (It can be assumed that L is sufficiently large to label all the distinct types of regions, wires and vertices that appear in the diagram.) We will generally use the more attractive "type notation" of Figure 3 for these diagrams, rather than the bare "untyped notation" of Figure 6. In these n-diagrams, which we typically draw in a 2-dimensional projection, vertices correspond to labels of dimension n, wires correspond to labels of dimension $n - 1$, and regions correspond to labels of dimension at most $n - 2$.

D. Further zigzag constructions

Here we collect some further technical results on zigzags and zigzag maps, which will be used later.

There is an obvious way in which zigzags can be concatenated, by gluing their diagrams horizontally.

Definition 17 (Zigzag concatenation). In a category $\mathbf{C}$, given zigzags Z, Z' such that the last regular object of Z equals the first regular object of Z', their *concatenation* is the zigzag $Z \circ Z'$ of length $n^Z + n^{Z'}$, obtained by drawing Z to the left of Z' such that their last and first regular level respectively coincides. For any such Z, Z', given zigzag maps $f : Z \to Y$ and $f' : Z' \to Y'$, we can also concatenate f and g in a precisely analogous way, yielding $f \circ g : Z \circ Z' \to Y \circ Y'$.

These compositional properties are perhaps unsurprising, given that we will use zigzags as the foundation of our approach to associative n-categories. Also note that zigzag concatenation is strictly associative, a property that is inherited by the theory of associative n-categories for composition in all dimensions.

Functors on base categories extend to zigzag categories, and the zigzag construction as a whole extends to $\mathbf{Cat}$. We omit the proofs, which are straightforward.

Lemma 18 (Zigzag functors). *A functor $F : \mathbf{C} \to \mathbf{D}$ extends to a zigzag functor $Z_F : Z_{\mathbf{C}} \to Z_{\mathbf{D}}$, acting on objects by direct application to all objects and morphisms in the zigzag diagram, and on morphisms by direct application to the entire commutative diagram defining the zigzag map. Furthermore, if F is fully faithful, so is Z_F.*

Lemma 19. *The zigzag construction extends to a functor $Z : \mathbf{Cat} \to \mathbf{Cat}$, mapping categories to zigzag categories, and functors to zigzag functors.*

The equalities in step (3) of Definition 7 give a strong restriction on which zigzags can have maps between them. In particular, if $f : Z \to Z'$ is a zigzag map, then the first and last regular objects of Z and Z' must be the same. This yields a natural partition of $Z_{\mathbf{C}}$ into a disjoint union of full subcategories, as follows.

Definition 20 (Local zigzag category). Given a category $\mathbf{C}$ with chosen objects A, B, the *local zigzag category* $Z_{\mathbf{C}}(A, B)$ is the full subcategory of $Z_{\mathbf{C}}$ containing zigzags whose first regular object is A, and whose last regular object is B.
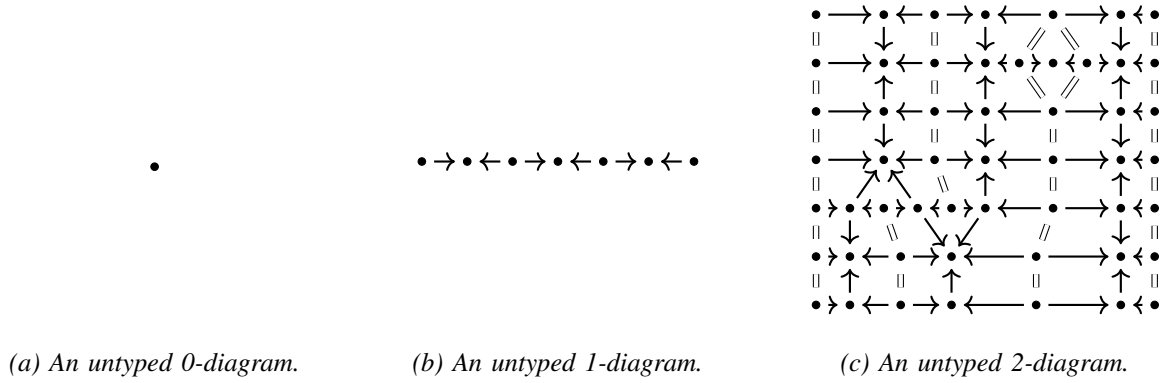
(a) An untyped 0-diagram. *(b) An untyped 1-diagram.* *(c) An untyped 2-diagram.*

Fig. 6: Examples of untyped diagrams.

Lemma 21 (Decomposition). $Z_\mathbf{C} = \coprod_{A,B \in \mathrm{Ob}(\mathbf{C})} Z_\mathbf{C}(A, B)$.

Zigzag maps are defined in terms of the construction of a commutative diagram. This construction is important, and we emphasize it with the following definition.

Definition 22. Given a zigzag map $f : Z \to Z'$, its *zigzag map diagram* is the corresponding diagram (as presented in Figure 5) with which it was defined.

This gives us a formal way to project diagrams in $Z_\mathbf{C}$ to give diagrams in $\mathbf{C}$, which we illustrate in Figure 7.

Definition 23. For a diagram $D : \mathbf{J} \to Z_\mathbf{C}$, its *deconstruction* $D^* : \mathbf{J}^* \to \mathbf{C}$ is the diagram obtained by taking the union of the diagrams of the zigzag maps $D(f)$ for all $f \in \mathrm{Mor}(\mathbf{J})$.

More precisely, the objects of the deconstructed diagram category $\mathbf{J}^*$ are given by a choice of $j \in \mathrm{Ob}(\mathbf{J})$, and a choice of a regular or singular height of $D(j)$; we write such an object as $\left(j, r_i^{D(j)}\right)$ or $\left(j, s_i^{D(j)}\right)$, where $r_i^{D(j)} \in D(j)_{\mathrm{reg}}$ and $s_i^{D(j)} \in D(j)_{\mathrm{sing}}$. The morphisms of $\mathbf{J}^*$ are given by adding for all $j \in \mathrm{Ob}(\mathbf{J})$ and $i \in D(j)_{\mathrm{sing}}$ morphisms

$\left(j, r_i^{D(j)}\right) \to \left(j, s_i^{D(j)}\right) \leftarrow \left(j, r_{i+1}^{D(j)}\right)$, and for all $f \in \mathrm{Mor}(\mathbf{J})$ with $f : j \to j'$, additional morphisms between singular heights $\left(j, s_i^{D(j)}\right) \to \left(j', s_{D(f)_{\mathrm{sing}}(i)}^{D(j')}\right)$ and regular heights $\left(j', r_i^{D(j')}\right) \to \left(j, r_{D(f)_{\mathrm{reg}}(i)}^{D(j)}\right)$.

Given a zigzag map $f : Z \to Z'$, we can restrict it to some contiguous subset of Z'_{reg}. We illustrate this idea in Figure 8, and develop it formally as follows. Here and throughout, the function $f_{\mathrm{reg}} : Z'_{\mathrm{reg}} \to Z_{\mathrm{reg}}$ is understood to act on pairs $a, b \in Z'_{\mathrm{reg}}$ elementwise.
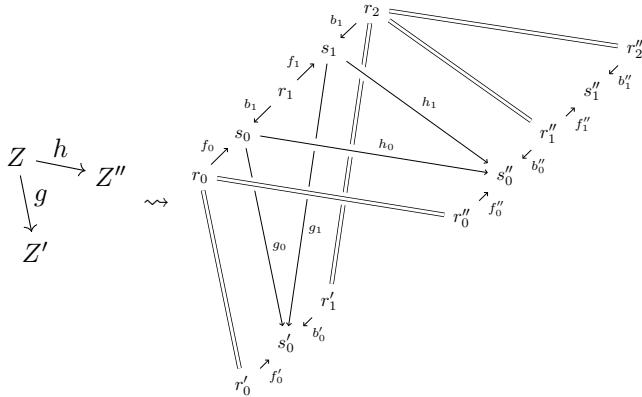
Definition 24 (Zigzag restriction). For a zigzag Z and a pair $a, b \in Z_{\mathrm{reg}}$ with $a \leqslant b$, the *restricted zigzag* $Z_{(a,b)}$ is that part of the zigzag diagram for Z that includes the regular objects r_a and r_b, and everything in between.

Definition 25 (Zigzag map restriction). For a zigzag map $f : Z \to Z'$ and $a, b \in Z'_{\mathrm{reg}}$ with $a < b$, the *restricted zigzag map* $f_{(a,b)} : Z_{f_{\mathrm{reg}}(a,b)} \to Z'_{(a,b)}$ is that part of the zigzag map diagram for f that includes the zigzag diagrams for $Z_{f_{\mathrm{reg}}(a,b)}$ and $Z'_{(a,b)}$, and the morphisms going between these parts.

III. CONTRACTION

We define contraction as follows.

Definition 26. Given a zigzag in $\mathbf{C}$, we define its *contraction* to be the zigzag of length 1 arising from the colimit in $\mathbf{C}$, if

Fig. 7: The deconstruction of a diagram in $Z_\mathbf{C}$, given as a diagram in $\mathbf{C}$.

Fig. 8: The restriction of a zigzag map $f : Z \to Z'$ (the entire diagram) to the regular heights $1, 2 \in Z'_{\mathrm{reg}}$, yielding $f_{(1,2)} : Z_{(1,3)} \to Z'_{(1,2)}$, (drawn in black.)

it exists, of its zigzag diagram:

$$(2)$$
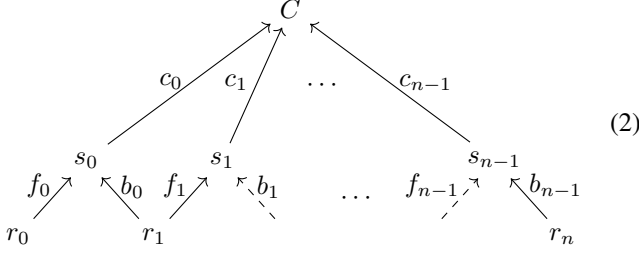
Given the structure of a zigzag diagram, we of course only need to define the cocone maps for the singular objects. If the colimit exists, then the contraction is defined to be the following zigzag in $\mathbf{C}$ of length 1:

$$(3)$$
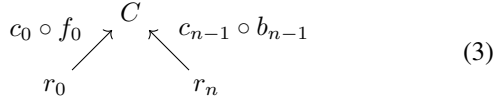
If the colimit does not exist, then the contraction is not defined. For our intended application this will frequently be the case, as the categories $Z_{\mathbf{L}}^n$ that we will be working with lack many colimits. We can interpret this as saying that contraction is nontrivial, and not always possible for an n-diagram.

Remark 27. In such a zigzag colimit diagram, note that the first and last regular objects r_0 and r_n, and their associated morphisms f_0 and b_{n-1}, do not affect the colimit. When it simplifies the narrative to do so, we will ignore them in our formal developments below.

A. Constructing zigzag colimits

Given a connected diagram in $Z_{\mathbf{C}}$, we build its colimit, or detect that such a colimit does not exist, by the following scheme. This scheme, and its correctness proofs, are the main mathematical contributions of this paper. Note that we do not assume that $\mathbf{C}$ itself has any particular colimits; but if $\mathbf{C}$ has few colimits, then the same will be true for $Z_{\mathbf{C}}$.

Definition 28 (Zigzag colimit)**.** For a category $\mathbf{C}$ with a terminal object, given a non-empty connected diagram $D : J \to Z_{\mathbf{C}}$, we build its colimit, or fail, according to the following scheme. To fix notation, we write C for the final colimit zigzag that we are trying to construct, and for each $j \in \mathrm{Ob}(\mathbf{J})$, we write $f^j : S_{\mathbf{C}} D(j) \to C$ for the corresponding cocone zigzag map.

(1) Build the diagram $\mathbf{J} \xrightarrow{D} Z_{\mathbf{C}} \xrightarrow{S_{\mathbf{C}}} \Delta$, and obtain its colimit. If no colimit exists, fail.
(2) Otherwise, we have a colimit object $c \in \mathrm{Ob}(\Delta)$, and cocone monotone functions $c_j : S_{\mathbf{C}} D(j) \to c$ for every $j \in \mathrm{Ob}(\mathbf{J})$.
(3) We choose the zigzag C to have length c, and we choose the monotone functions $(f^j)_{\mathrm{sing}} = c_j$.
(4) We now perform the following subconstruction for each $k \in [c]$, as follows.
   (i) Restrict the diagram $D : \mathbf{J} \to Z_{\mathbf{C}}$ to a diagram $D_k : \mathbf{J} \to Z_{\mathbf{C}}$, by defining $D_k(j)$ on an object

$j \in \mathrm{Ob}(\mathbf{J})$ as the restricted zigzag $D(j)_{c'_j(k,k+1)}$, and similarly on morphisms.[5]
   (ii) Build the deconstruction $(D_k)^* : \mathbf{J}^* \to \mathbf{C}$,[6] and obtain its colimit. If no colimit exists, fail.
   (iii) Otherwise, we have a colimit object $p \in \mathrm{Ob}(\mathbf{C})$, and for any $j \in \mathrm{Ob}(\mathbf{J})$ and $i \in D(j)_{\mathrm{sing}}$, a cocone morphism of type $p_i^j : (D_k)^*(j, s_i^{D(j)}) \to p$.
   (iv) Build a zigzag C_k of height 1 as follows. Choose some $j \in \mathrm{Ob}(\mathbf{J})$ with $D_k(j)_{\mathrm{sing}} = m > 0$.[7] Define the forward map as $f_0^{C_k} = p_0^j \circ f_0^{D_k(j)}$, and the backward map as $b_0^{C_k} = p_{m-1}^j \circ b_{m-1}^{D_k(j)}$. Hence obtain $r_0^{C_k}$ and $r_1^{C_k}$ as the sources of $f_0^{C_k}$ and $b_0^{C_k}$ respectively. We set $s_0^{P_k} = p$.
   (v) For a fixed $j \in \mathrm{Ob}(\mathbf{J})$, build a zigzag map of type $f^{j,k} : D_k(j) \to C_k$ by choosing the monotone map as the unique one of type $D_k(j)_{\mathrm{sing}} \to [1]$, and by choosing the singular morphisms at source singular height i as p_i^j.
(5) Build the colimit zigzag C as the concatenation of the length-1 zigzags C_k.[8]
(6) For each value of j, build the cocone zigzag map f^j as the concatenation of the zigzag maps $f^{j,k}$ for $k \in [c]$.

This completes the description of the colimit construction scheme. The correctness proofs follow in Section III-C.

We illustrate this procedure in Figure 9, which shows the computation of a pushout in $Z_{\mathbf{C}}$. The top-left, bottom-left and top-right zigzags are given, as well as the maps between them. The length of the bottom-right zigzag, and its incoming monotone maps, are determined by taking a pushout in Δ. The regular objects of the bottom-right zigzag are completely

---

[5]Recall Definition 5 of $(-)' : \Delta \to \Delta_{\cong}^{\mathrm{op}}$.
[6]Recall Definition 23 of the deconstruction procedure.
[7]Such a j must exist, since a colimit in Δ of empty sets is empty.
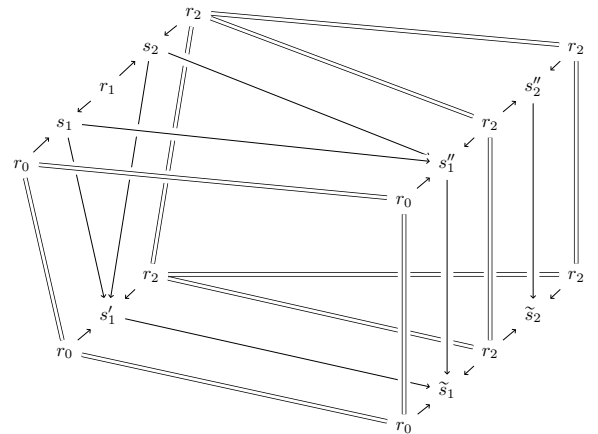[8]Recall Definition 17 of concatenation of zigzags and their maps.

Fig. 9: A pushout in $Z_{\mathbf{C}}$.

determined by the incoming maps, and the singular objects are computed as colimits over the 'incoming diagrams':

$$\widetilde{s}_1 = \mathrm{colim} \left(\begin{array}{c} \xymatrix{ & s_2 & \\ s_1 \ar[ur]^{r_1} \ar[rr] \ar[dr] & & s_1'' \\ & s_1' & } \end{array} \right) \qquad \widetilde{s}_2 = s_2''$$

The morphisms into the singular objects are given by the obvious morphisms into the colimits.

In the implementation *homotopy.io*, this colimit construction scheme provides the main recursive algorithm for performing contractions of typed diagrams, as objects of $Z_{\mathbf{L}}^n$. While we do not go into detail regarding the implementation, it is at least worth noting that termination is clear, since colimits in the base category $\mathbf{L}$ can be trivially computed, and for a finite diagram, this colimit construction scheme involves only finitely many loops, with all recursion being to strictly lower-dimensional instances.

B. Examples

We already encountered some nontrivial examples of contractions, in Figures 1 and 2. We give some further examples here. In the online versions of the proofs, you can view the contraction yourself by changing the setting of the "Slice" control in the top-right, or perform the contraction yourself (where possible), using the mouse to drag one of the vertices vertically towards the centre of the diagram.

Example 29 (Link to online proof). Here we perform a contraction in $Z_{\mathbf{L}}^2$ of a zigzag of length 2, containing 2 vertices. In the contracted diagram, these vertices are at the same height.

$$\rightsquigarrow \tag{4}$$
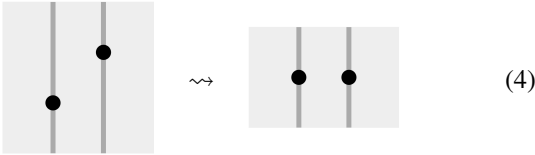
Example 30 (Link to online proof). In this non-example, again in $Z_{\mathbf{L}}^2$, the colimit construction procedure fails at step (1), since the diagram $\mathbf{J} \xrightarrow{D} Z_{\mathbf{C}} \xrightarrow{S_{\mathbf{C}}} \Delta$ has image $[1] \leftarrow [0] \rightarrow [1]$, which does not have a colimit:

$$\tag{5}$$

To understand why this contraction does not exist, consider that, if it could be constructed, the resulting unique singular height would have to contain 2 vertices, with one to the left of the other, as follows:

$$\tag{6}$$

However, the colimit construction algorithm has no way to "break the symmetry", and cannot proceed. The implementation *homotopy.io* uses some additional techniques which allow us to break the symmetry here; in the online proof, we apply these techniques by dragging the upper vertex of (5) in a south-east or south-west direction, to produce the two images given in (6).

Example 31 (Link to online proof). If we modify the previous example by putting a wire in between the vertices, the diagram will now contract successfully:

$$\rightsquigarrow \tag{7}$$

This is because the colimit diagram in Δ now has the image $[2] \xleftarrow{0 \to 1} [1] \xrightarrow{0 \to 0} [2]$, which does have a colimit.

Suppose that we are taking the contraction of a typed n-diagram D—that is, an object of $Z_{\mathbf{L}}^n$—which is well-typed with respect to some signature Σ (see Section II-C for a brief discussion of type checking.) Even if the contraction of D exists, yielding a new object D' of $Z_{\mathbf{L}}^n$, it does not follow that D' will again be well-typed with respect to Σ; the entire contraction $D \to D'$ must be passed through the type checker to verify this. We show such an example here.

Example 32 (Link to online proof). In this example, again in $Z_{\mathbf{L}}^2$, we contract a zigzag of length 2, as follows:

$$\rightsquigarrow \tag{8}$$

Here we "fuse" two endomorphisms on a wire into a single endomorphism, with the colimit construction procedure successfully returning the right-hand diagram. Both of these diagrams type check, but the contraction process as a whole does not, because homotopies may only "move" parts of the diagram around, not change the structure of individual labels. As a result, in the online proof, clicking and dragging either of the two vertices will have no effect, as the contraction above will be silently blocked by the type checker. This shows the way that contraction and type checking interact in the implementation.

C. Correctness

Theorem 33. *Let $\mathbf{C}$ be a category with a terminal object and let $D : \mathbf{J} \to Z_{\mathbf{C}}$ be a non-empty connected diagram. Then, D has a colimit if and only if the procedure in Definition 28 succeeds (that is, if the colimits in step (1) and (4.ii) exist), and the procedure constructs it.*

Remark 34. Since the category $Z_{\mathbf{C}}$ is a disjoint union of local zigzag categories $Z_{\mathbf{C}}(a, b)$ for objects a, b in $\mathbf{C}$ (see

Definition 20), and since Theorem 33 applies to connected non-empty diagrams, it is also true as stated for the categories $Z_{\mathbf{C}}(a,b)$ replacing $Z_{\mathbf{C}}$. Moreover, note that if $\mathbf{C}$ is a category with a terminal object $*$, then $Z_{\mathbf{C}}(a,b)$ has a terminal object (namely, the zigzag $a \to * \leftarrow b$). Since all categories of diagrams may be obtained as iterated local zigzag categories, Theorem 33 holds for such categories.

We prove Theorem 33 in two steps. First, we show that if the colimits in step (1) and (4.ii) exist, then the constructed cocone is indeed colimiting. Then, we prove that if a colimit of a diagram $D : J \to Z_{\mathbf{C}}$ exist, then the colimits in step (1) and (4.ii) must also exist.

D. The procedure correctly computes colimits

We prove the first part of Theorem 33: If the colimits in step (1) and (4.ii) exist, then the constructed cocone is indeed a colimiting cocone of the diagram $D : J \to Z_{\mathbf{C}}$.

The proof boils down to the following categorical fact: given a (Grothendieck) opfibration $F : \mathbf{A} \to \mathbf{B}$, then colimits in $\mathbf{A}$ can be computed in terms of colimits in $\mathbf{B}$ and in the fibre categories $F^{-1}(b)$ for objects $b \in \mathbf{B}$.

Opfibrations and colimits. We recall the following terminology. Given a functor $F : \mathbf{A} \to \mathbf{B}$, a morphism $\phi : a \to a'$ in $\mathbf{A}$ is called *opcartesian* if for any morphism $\psi : a \to a''$ in $\mathbf{A}$ and $g : F(a') \to F(a'')$ in $\mathbf{B}$ such that $g \circ F(\phi) = F(\psi)$, there exists a unique $\chi : a' \to a''$ such that $\chi \circ \phi = \psi$ and $F(\chi) = g$. A functor $F : \mathbf{A} \to \mathbf{B}$ is an *opfibration* if for any $a \in \mathbf{A}$ and $h : F(a) \to b$ in $\mathbf{B}$, there is a opcartesian morphism $\phi : a \to a'$ with $F(\phi) = h$. For an opfibration $F : \mathbf{A} \to \mathbf{B}$ and an object $b \in \mathbf{B}$, the *fibre category* $F^{-1}(b)$ is the subcategory of $\mathbf{A}$ with objects and morphisms mapping to b and id_b, respectively. Given a morphism $\sigma : b \to b'$ in $\mathbf{B}$, the *base change functor* $\sigma_* : F^{-1}(b) \to F^{-1}(b')$ maps an object a in the fibre over b to the codomain of the opcartesian morphism lifting $\sigma : Fa \to b'$ and a morphism $f : a \to a'$ over id_b to the morphism $\sigma_* a \to \sigma_* a'$ obtained from opcartesianity of the lift of $\sigma : Fa \to b'$.

We recall the following basic fact about opfibrations.

Proposition 35. *Let $F : \mathbf{A} \to \mathbf{B}$ be an opfibration and let $D : \mathbf{J} \to \mathbf{A}$ be a diagram such that FD has a colimit. If all fibres have $\mathbf{J}$-colimits and the base change functor $\sigma_* : F^{-1}(b) \to F^{-1}(b')$ preserves them for all $\sigma : b \to b'$ in $\mathbf{B}$, then D has a colimit and F preserves it.*

This proposition is proven later as Proposition 42.

Explicitly, we can compute this colimit in terms of the colimit of $FD : \mathbf{J} \to \mathbf{B}$ as follows: Lift the universal cocone morphisms $\lambda^j : FD_j \to \mathrm{colim}FD$ to opcartesian morphisms $\phi^j : D_j \to \lambda^j_*(D_j)$, where $F\left(\lambda^j_*(D_j)\right) = \mathrm{colim}FD$. Opcartesianity of ϕ^j gives rise to morphisms $\lambda_\sigma : \lambda^j_*(D_j) \to \lambda^{j'}_*(D_{j'})$ for $\sigma : j \to j'$ in $\mathbf{J}$, making this into a diagram $\mathbf{J} \to F^{-1}(\mathrm{colim}FD)$. A colimiting cocone $\mu^j : \lambda^j_*(D_j) \to X$ of this diagram $\mathbf{J} \to F^{-1}(\mathrm{colim}FD)$ induces a colimiting cocone $\mu^j \circ \phi^j : D_j \to X$ of D.

$S_{\mathbf{C}}$ is an opfibration for cocomplete $\mathbf{C}$. Given a zigzag Z (drawn on the left) with a chosen regular object (here labelled r), we define a new zigzag $\widetilde{Z}$ (drawn on the right) in which the regular object is 'expanded' into two regular objects, and a morphism of zigzags $Z \to \widetilde{Z}$ as follows:
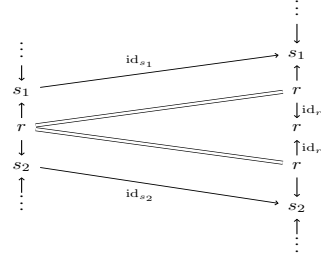
If $\mathbf{C}$ is cocomplete, and Z is a zigzag with a chosen pair of adjacent singular objects (here labelled s_1 and s_2 on the left), we define a new zigzag $\widetilde{Z}$ in which the singular objects are 'collapsed' into a single singular object, given by the pushout of s_1 and s_2 over the intermediate regular object, and a morphism $Z \to \widetilde{Z}$:
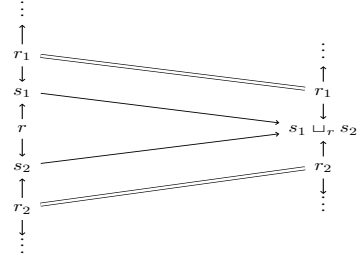
Given a zigzag Z and a monotone map $h : Z_{\mathrm{sing}} \to I$ into some finite totally ordered set I, we iterate these operations to produce a zigzag $\widetilde{Z}$ of length $|I|$ and a morphism of zigzags $\widetilde{h} : Z \to \widetilde{Z}$ with underlying monotone map $\widetilde{h}_{\mathrm{sing}} = h$ as illustrated in the following example lifting the constant monotone map $\{1 < 2 < 3\} \to \{1 < 2 < 3\}$, $x \mapsto 1$:
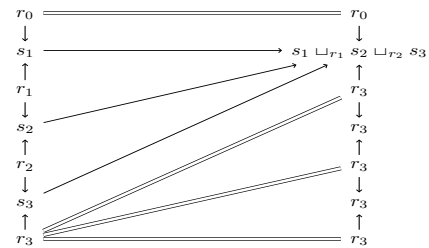
Here, the left zigzag and the underlying monotone map are given; the right zigzag and the map of zigzags are produced by 'expanding' and 'collapsing'. This ability to 'lift' monotone maps $h : Z_{\mathrm{sing}} \to I$ to maps of zigzags leads to the following proposition.

Proposition 36. *If $\mathbf{C}$ is cocomplete, then the singular monotone functor $S_{\mathbf{C}} : Z_{\mathbf{C}} \to \Delta$ is an opfibration.*

Proof. Given a zigzag Z and a monotone map $h : Z_{\mathrm{sing}} \to I$ into some totally ordered set I, we lift h to a map of zigzags $\widetilde{h} : Z \to \widetilde{Z}$ obtained by 'collapsing' and 'expanding' Z, as

9

described above. The fact that $\widetilde{h}$ is opcartesian corresponds precisely to the universal property of the colimits in the collapse operation. $\qquad\square$

For a finite totally ordered set I, the fibre category $S_{\mathbf{C}}^{-1}(I)$ is the category of zigzags of length $|I|$ with morphisms the maps of zigzags whose underlying monotone map is the identity. Explicitly, this category is the disjoint union

$$\bigsqcup_{r_i \in \mathrm{ob}\,\mathbf{C} \text{ for } i \in I} \quad \bigtimes_{0 \leqslant i \leqslant |I|-1} (r_i, r_{i+1})/\mathbf{C},$$

where $(a,b)/\mathbf{C}$ denotes the over-category whose objects are pairs of morphisms $(a \to x, b \to x)$ and morphisms $(a \to x, b \to x) \to (a \to y, b \to y)$ are morphisms $x \to y$ making the obvious triangles commute.

Given a monotone map $\lambda : I \to J$, the induced base change functor $S_{\mathbf{C}}^{-1}(I) \to S_{\mathbf{C}}^{-1}(J)$ maps a zigzag of length $|I|$ to a zigzag of length $|J|$ by expanding and collapsing according to the monotone map λ.

Corollary 37. *Let $\mathbf{C}$ be cocomplete and let $D : \mathbf{J} \to Z_{\mathbf{C}}$ be a connected, non-empty diagram such that $S_{\mathbf{C}}D$ has a colimit. Then, D has a colimit C, which is preserved by $S_{\mathbf{C}}$, and which can be explicitly constructed as follows:*

1) *Construct a colimit C_{sing} of $S_{\mathbf{C}}D$ with colimiting cocone $f_{\mathrm{sing}}^j : D_{\mathrm{sing}}^j \to C_{\mathrm{sing}}$.*
2) *For every $j \in \mathbf{J}$, 'expand' and 'collapse' the zigzag D^j to a zigzag $\widetilde{D}^j$ of length $|C_{\mathrm{sing}}|$ according to the monotone map $f_{\mathrm{sing}}^j : D_{\mathrm{sing}}^j \to C_{\mathrm{sing}}$. This gives rise to a diagram $\widetilde{D} : \mathbf{J} \to Z_{\mathbf{C}}$ in which every zigzag has the same length and every morphism of zigzags has underlying identity monotone map.*
3) *For every singular height $i \in C_{\mathrm{sing}}$, let s_i be the colimit in $\mathbf{C}$ over the diagram $\widetilde{D}|_i : \mathbf{J} \to \mathbf{C}$ obtained by restricting the diagram $\widetilde{D} : \mathbf{J} \to Z_{\mathbf{C}}$ to the singular objects at height i and the morphisms between them (recall that all maps of zigzags in the image of $\widetilde{D}$ have underlying identity monotone map).*
4) *For every regular height $i \in (C_{\mathrm{sing}})' = C_{\mathrm{reg}}$, define the regular object r_i to be equal to the regular object of $\widetilde{D}^j$ at height i for some (and hence any) $j \in \mathbf{J}$.*
5) *Define the forward and backward morphisms of C and the singular morphisms of $f^j : D^j \to C$ as the obvious morphisms into the colimits s_i.*

Proof. The fibre $S_{\mathbf{C}}^{-1}(I)$ has all connected colimits since connected colimits in over-categories can be constructed as colimits in the original category $\mathbf{C}$ in the obvious way. The base change functors can be factored into functors expanding a single regular object or collapsing a pair of adjacent singular objects. Explicitly, the corresponding base change functors are of the form

$$\cdots \times \mathrm{id}_{(r_{i-1}, r_i)/\mathbf{C}} \times r_i \times \mathrm{id}_{(r_i, r_{i+1})/\mathbf{C}} \times \cdots,$$

where $r_i : * \to (r_i, r_i)/\mathbf{C}$ picks out the object $r_i \xrightarrow{\mathrm{id}_{r_i}} r_i \xleftarrow{\mathrm{id}_{r_i}} r_i$, and

$$\cdots \times \mathrm{id}_{(r_{i-2}, r_{i-1})/\mathbf{C}} \times (- \sqcup_{r_i} -) \times \mathrm{id}_{(r_{i+1}, r_{i+2})/\mathbf{C}} \times \cdots,$$

where $- \sqcup_{r_i} - : (r_{i-1}, r_i)/\mathbf{C} \times (r_i, r_{i+1})/\mathbf{C} \to (r_{i-1}, r_{i+1})/\mathbf{C}$ takes the pushout of the inner span. It is clear that both functors preserve connected, non-empty colimits.

It therefore follows from Proposition 35 that D has a colimit which is preserved by $S_{\mathbf{C}}$ and is constructed as described. $\quad\square$

Colimits in $S_{\mathbf{C}}$ if $\mathbf{C}$ is not cocomplete. Categories of typed or untyped diagrams—such as the category $\Delta = Z_{\mathbf{1}}$, or iterated zigzag categories on Δ—are far from cocomplete. In particular, Corollary 37 does not hold in this setting.

Recall that we have 'collapsed' singular objects by taking a colimit in $\mathbf{C}$, and have later again taken colimits in $\mathbf{C}$ to compute the colimit of the diagram in the fiber. In other words, we have computed the colimit of $\mathbf{J} \to Z_{\mathbf{C}}$ by first computing the colimit C_{sing} in Δ and then, for every $i \in C_{\mathrm{sing}}$, taking several consecutive colimits in $\mathbf{C}$. If $\mathbf{C}$ is not cocomplete, it is possible that some of these intermediate colimits do not exist, even if the overall colimit does exist. We can avoid this issue by only 'formally' taking intermediate colimits. This can be formalized by passing to the free completion of $\mathbf{C}$, as follows.

Let $y : \mathbf{C} \to \widehat{\mathbf{C}} := [\mathbf{C}, \mathrm{Set}]^{\mathrm{op}}$ denote the 'dual' Yoneda embedding of $\mathbf{C}$. The functor y has the convenient property that it preserves and reflects all colimits; in particular, a diagram $D : \mathbf{J} \to \mathbf{C}$ has a colimit if and only if the colimit of the diagram $yD : \mathbf{J} \to \widehat{\mathbf{C}}$ is representable (that is, is in the essential image of y.) Moreover, y gives rise to a fully faithful functor $Z_y : Z_{\mathbf{C}} \to Z_{\widehat{\mathbf{C}}}$.

Proposition 38. *Let $D : \mathbf{J} \to Z_{\mathbf{C}}$ be a connected, non-empty diagram such that the colimits in step (1) and (4.ii) of Definition 28 exist. Then the cocone constructed in Definition 28 is colimiting.*

Proof. It follows from the existence of the colimit in step (1) and Corollary 37 that the composite $\mathbf{J} \to Z_{\mathbf{C}} \hookrightarrow Z_{\widehat{\mathbf{C}}}$ has a colimiting cocone, constructed as in Corollary 37. The existence of the colimits in step (4.ii) of Theorem 33 imply that the singular objects of the constructed zigzag (constructed in step 3 of Corollary 37) are representable. Hence, the constructed cocone is in the image of the fully faithful $Z_y : Z_{\mathbf{C}} \to Z_{\widehat{\mathbf{C}}}$, and is therefore a colimit of $\mathbf{J} \to Z_{\mathbf{C}}$. $\quad\square$

E. The procedure detects all colimits

We now prove the second part of Theorem 33: if a connected, non-empty diagram $D : \mathbf{J} \to Z_{\mathbf{C}}$ has a colimit, then the colimits in step (1) and (4.ii) of Definition 28 exist.

Proposition 39. *Let $\mathbf{C}$ be a category with a terminal object. The functor $S_{\mathbf{C}} : Z_{\mathbf{C}} \to \Delta$ preserves connected colimits.*

Proof. Given a set X, we define $\Delta_=(X)$ as the following generalization of the category $\Delta_=$ from Definition 4: its objects are pairs (O, f) of a non-empty totally ordered set O and a function $f : O \to X$, and its morphisms $(O, f) \to (O', f')$ are regular monotone maps $\rho : O \to O'$ such that $f' \circ \rho = f$. Note that $\Delta_=(X)$ is the comma category F/X, where $F : \Delta_= \to \mathrm{Set}$ is the forgetful functor.

The regular monotone functor $R_{\mathbf{C}} : Z_{\mathbf{C}} \to (\Delta_=)^{\mathrm{op}}$ factors through a functor $\mathbb{L} : Z_{\mathbf{C}} \to (\Delta_=(\mathrm{ob}\,\mathbf{C}))^{\mathrm{op}}$ mapping a zigzag

Z to its totally ordered set of regular objects Z_{reg} together with the function $Z_{\text{reg}} \to \text{ob}\mathbf{C}$, $i \mapsto r_i$. We construct a right adjoint $\mathbb{R} : (\Delta_=(\text{ob}\mathbf{C}))^{op} \to Z_{\mathbf{C}}$ as follows. The functor $\mathbb{R}$ maps an object (O, f) to the zigzag of length $|O|$ with regular objects determined by f and with singular objects given by the terminal object of $\mathbf{C}$. It maps a morphism $\lambda : (O, f) \to (O', f')$ to the unique morphism of zigzag with underlying regular monotone map λ. The natural transformation

$$\text{Hom}_{Z_{\mathbf{C}}}(Z, \mathbb{R}(O, f)) \to \text{Hom}_{\Delta_=(\text{ob}\mathbf{C})}((O, f), \mathbb{L}A)$$

mapping a map of zigzags $Z \to \mathbb{R}(O, f)$ to its underlying regular map is a natural isomorphism. Hence, $\mathbb{R}$ is right adjoint to $\mathbb{L}$ and in particular, $\mathbb{L} : Z_{\mathbf{C}} \to (\Delta_=(\text{ob}\mathbf{C}))^{op}$ preserves colimits.

Therefore, to show that $S_{\mathbf{C}} : Z_{\mathbf{C}} \to \Delta$ preserves connected colimits, it suffices to show that the composite functor

$$(\Delta_=(\text{ob}\mathbf{C}))^{op} \to (\Delta_=)^{op} \xrightarrow{(-)'} \Delta$$

preserves connected colimits. Since $(-)'$ is an equivalence, it suffices to show that $\Delta_=(\text{ob}\mathbf{C}) \to \Delta_=$ preserves connected limits. Since $\Delta_=(\text{ob}\mathbf{C}) = F/X$ is a comma category, this follows from Proposition 43. $\qquad\square$

Corollary 40. *Let $\mathbf{C}$ be a category with a terminal object. The functor $Z_{\mathbf{C}} \to Z_{\widehat{\mathbf{C}}}$ preserves connected, non-empty colimits.*

Proof. Let $D : \mathbf{J} \to Z_{\mathbf{C}}$ be a connected non-empty diagram, and let $a^j : D^j \to C$ be a colimiting cocone. By Proposition 39, the cocone $c^j_{\text{sing}} : D^j_{\text{sing}} \to C_{\text{sing}}$ is colimiting in Δ. By Corollary 37, the composite $\mathbf{J} \to Z_{\mathbf{C}} \to Z_{\widehat{\mathbf{C}}}$ has a colimiting cocone $\widehat{c}^j : D^j \to \widehat{C}$. In particular, there is a morphism of cocones $\mu : \widehat{C} \to C$ in $Z_{\widehat{C}}$. In the following, we show that μ is an isomorphism.

Since $\mu_{\text{sing}} : \widehat{C}_{\text{sing}} \to C_{\text{sing}}$ is a morphism of cocones $\left(D^j_{\text{sing}} \to \widehat{C}_{\text{sing}}\right) \to \left(D^j_{\text{sing}} \to C_{\text{sing}}\right)$, and since both cocones are colimiting, it follows that μ_{sing} is the identity and that $c^j_{\text{sing}} = \widehat{c}^j_{\text{sing}}$. Denote the regular objects of C by $r_0, \ldots, r_n$. Since there is a morphism $\mu : \widehat{C} \to C$ with $\mu_{\text{sing}} = \text{id}$, it follows that the regular objects of $\widehat{C}$ are also $r_0, \ldots, r_n$. In particular, the morphism μ can be understood as a morphism in the category $\widehat{\mathbf{E}} := \times_i (r_i, r_{i+1})/\widehat{\mathbf{C}}$. Denoting $\mathbf{E} := \times_i (r_i, r_{i+1})/\mathbf{C}$, we observe that the obvious functor $\widehat{\mathbf{E}} \to [\mathbf{E}, \text{Set}]^{op}$ is an equivalence.

Let E be an object of $\mathbf{E}$—or equivalently, a zigzag with regular objects $r_0, \ldots, r_n$ and singular objects in $\mathbf{C}$—and let $\lambda : \widehat{C} \to E$ be a morphism in $\widehat{\mathbf{E}}$. Then, the composite $\lambda \circ \widehat{c}^j : D^j \to E$ is a cocone of $\mathbf{J} \to Z_{\mathbf{C}}$. In particular, there is a unique morphism $\phi : C \to E$ in $Z_{\mathbf{C}}$ such that $\lambda \circ \widehat{c}^j = \phi \circ c^j = \phi \circ \mu \circ \widehat{c}^j$, or equivalently such that $\lambda = \phi \circ \mu$. Applying the singular monotone functor $S_{\mathbf{C}}$ and using that $\lambda_{\text{sing}} = \mu_{\text{sing}} = \text{id}$, it follows that $\phi_{\text{sing}} = \text{id}$. We can therefore summarize the preceding paragraph as follows: given an object E of $\mathbf{E}$ and a morphism $\lambda : \widehat{C} \to E$ in $\widehat{\mathbf{E}}$, there is a unique morphism $\phi : C \to E$ in $\mathbf{E}$ such that $\lambda = \phi \circ \mu$. Since $\widehat{\mathbf{E}}$ is equivalent to the free completion $[\mathbf{E}, \text{Set}]^{op}$ of $\mathbf{E}$, this means that $\widehat{C}$ is in the essential image of $\mathbf{E} \to \widehat{\mathbf{E}}$ and hence isomorphic to C. $\qquad\square$

Corollary 41. *Let $\mathbf{C}$ be a category with a terminal object and let $\mathbf{J} \to Z_{\mathbf{C}}$ be a connected, non-empty diagram admitting a colimit. Then, the colimits in step (1) and (4.ii). of Definition 28 exist.*

Proof. The colimit in step (1) exists since the singular monotone functor $S_{\mathbf{C}} : Z_{\mathbf{C}} \to \Delta$ preserves connected, non-empty colimits (Proposition 39.) The existence of the colimit in step (4.ii) is equivalent to the representability of the singular objects in step 3 of Corollary 37. This follows since $Z_{\mathbf{C}} \to Z_{\widehat{\mathbf{C}}}$ preserves connected, non-empty colimits (Corollary 40.) $\qquad\square$

We can now give a proof of Theorem 33.

Proof of Theorem 33. Proposition 38 asserts that if the procedure succeeds, then the cocone constructed in Definition 28 is colimiting. Conversely, Corollary 41 shows that if $D : \mathbf{J} \to Z_{\mathbf{C}}$ has a colimit, then the procedure succeeds. $\qquad\square$

F. Additional results

Here we state without proof two technical results which we rely on for the above development.

Here we prove two categorical results used in the main paper, which we expect to be known to experts.

Proposition 42. *Let $F : \mathbf{A} \to \mathbf{B}$ be a Grothendieck opfibration and let $D : \mathbf{J} \to \mathbf{A}$ be a diagram such that FD has a colimit. If all fibres have $\mathbf{J}$-colimits and the base change functor $\sigma_* : F^{-1}(b) \to F^{-1}(b')$ preserves them for all $\sigma : b \to b'$ in $\mathbf{B}$, then D has a colimit and F preserves it.*

Proposition 43. *Let $F : \mathbf{C} \to \mathbf{D}$ be a functor and let X be an object of $\mathbf{D}$. Then, the forgetful functor $F/X \to \mathbf{C}$ from the comma category into $\mathbf{C}$ preserves connected limits.*
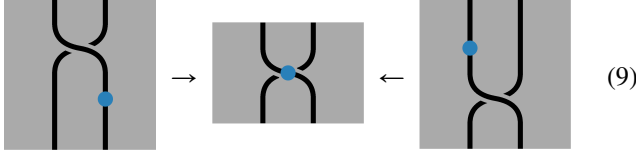
IV. HOMOTOPY CONSTRUCTION

Here we show that contraction can be used as a general technique to construct nontrivial homotopies. In particular, we analyze the 4-dimensional "naturality" homotopy, and the 5-dimensional "naturality of naturality" homotopy. We first introduce some simple additional techniques, which are used together with contraction in the tool to produce these examples.

The examples come with direct links to the formalized proofs in the online proof assistant *homotopy.io*, where the interested reader can investigate them. To explore them, change the parameters of the "Slice" control at the top-right. You can also manipulate them directly; for example to execute a homotopy, use the mouse to drag a vertex (or a crossing) up or down, or drag a wire to the left or right. Further guidance on using the tool is available on the nLab [7].

As well as contraction, the tool makes use of some simple additional recursive methods for homotopy construction methods, which involve contraction or its opposite in a slice of the diagram.

A. Naturality (*Link to online proof*)

Here we build the following "naturality" homotopy, during which a vertex moves through a braiding, as the following zigzag of length 1 in $Z_{\mathbf{L}}^3$ (or equivalently, as an object of $Z_{\mathbf{L}}^4$):

 (9)

To construct this homotopy, we begin by following the steps illustrated in Figure 10, yielding a proof which is a length-5 zigzag in $Z_{\mathbf{L}}^3$. Each of these steps is obtained by contracting, or performing one of a limited range of related recursive methods on some slice of the diagram; for example, in the arrow labelled * we contract the entire diagram, and in the arrow labelled † we perform a contraction within the first regular height of the diagram. By projecting out an extra dimension, we can view the entire proof that we have constructed as a 2-dimensional graphic, giving information about the overall structure of our proof, as shown in the first image here:

 (10)
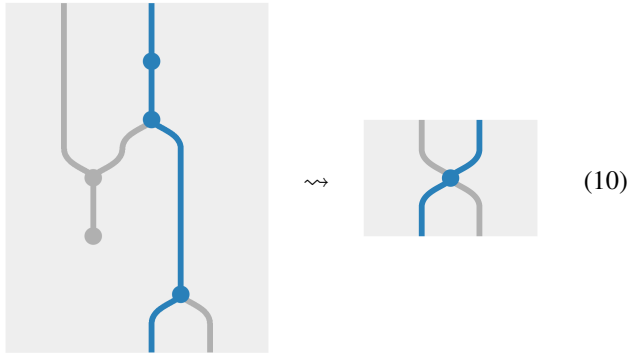
We then contract this, and this entire proof collapses to a zigzag of length 1, which performs the naturality move in a single step, shown in projection as the second diagram above. Viewing this as a "movie" gives back precisely the desired homotopy (9) above.

B. Naturality of naturality (*Link to online proof*)

This homotopy has the following 4-dimensional diagrams as its source and target respectively:

 (11)

 (12)
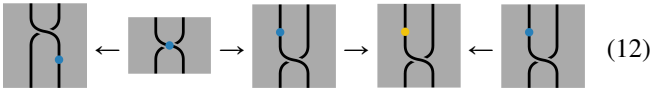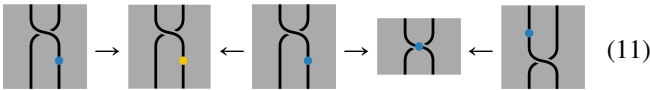
These diagrams feature a 3-cell drawn in blue, a crossing, and a 4-cell drawn in yellow, which acts as an endomorphism of the blue 3-cell. The source (11) describes a composite process that applies the 4-cell to the 3-cell, then pulls the 3-cell through the braiding; in the target (12), we instead first pull the 3-cell through the braiding, and then apply the 4-cell. These source

and target 4-diagrams are not homotopies, since they involve the yellow 4-cell, which is an algebraic move.

The "naturality of naturality" homotopy exhibits that the composites (11) and (12) are homotopic. The proof is constructed in *homotopy.io* as a zigzag of length 14 in $Z_{\mathbf{L}}^4$ (or alternatively, an object in $Z_{\mathbf{L}}^5$.) We build it by starting with the source 4-diagram (11), and manipulating it using our contraction-based methods. We give it as a movie (in which every frame shows a 4-dimensional diagram, viewed in 2-dimensional projection) in Figure 11.

Note that the first frame of this movie is given by the projection of the contracted naturality homotopy (the second image in (10)), composed with the yellow 4-cell, and the last frame has these same components composed a different way; the yellow 4-cell is indeed "pulled through the naturality" over the course of the proof, as we expect.

The proof as a whole has an interesting structure, which the movie of Figure 11 makes clear: we create a bubble, enlarge it, wrap it around the yellow 4-cell, and then contract the remaining parts, with the result being that the yellow 4-cell has moved to the other side of the naturality homotopy. Building this proof required repeated use of contraction, not only on the 4-dimensional term being manipulated as the proof was being developed, but also in 3-dimensional slices of those terms, which then propagated recursively to the entire 4-diagram.

As before, the entire 5-dimensional proof can be viewed as a single 2-dimensional projected image, by ignoring the lowest 3 dimensions. We represent it in the first image here:

 (13)
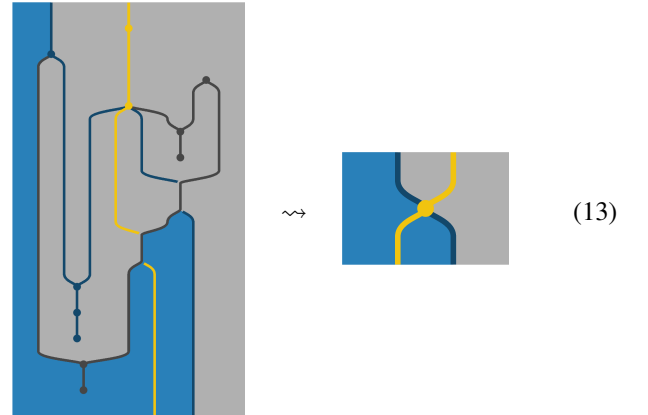
This entire proof contracts to a zigzag of length 1, and we give this contraction as the second image.

To summarize, we have shown how contraction can be used as the main workhorse for manipulating terms in an associative n-category, including the tasks of building an initial diagram, manipulating it (both at the top dimension and in lower dimensions) to obtain a proof object, and then contracting that proof object itself to yield a short witness for the logical statement being established.

REFERENCES

[1] C. Dorn, "Associative n-categories," Ph.D. dissertation, 2018, arXiv:1812.10586.

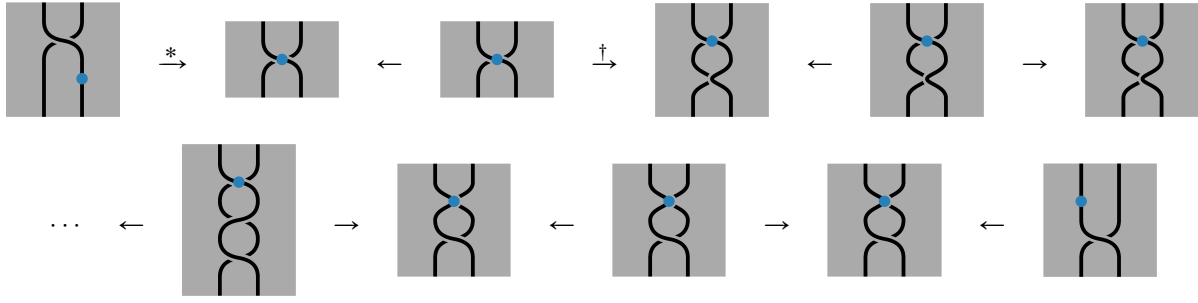[2] C. Dorn, C. Douglas, and J. Vicary, "The theory of associative n-categories," 2019, in preparation.

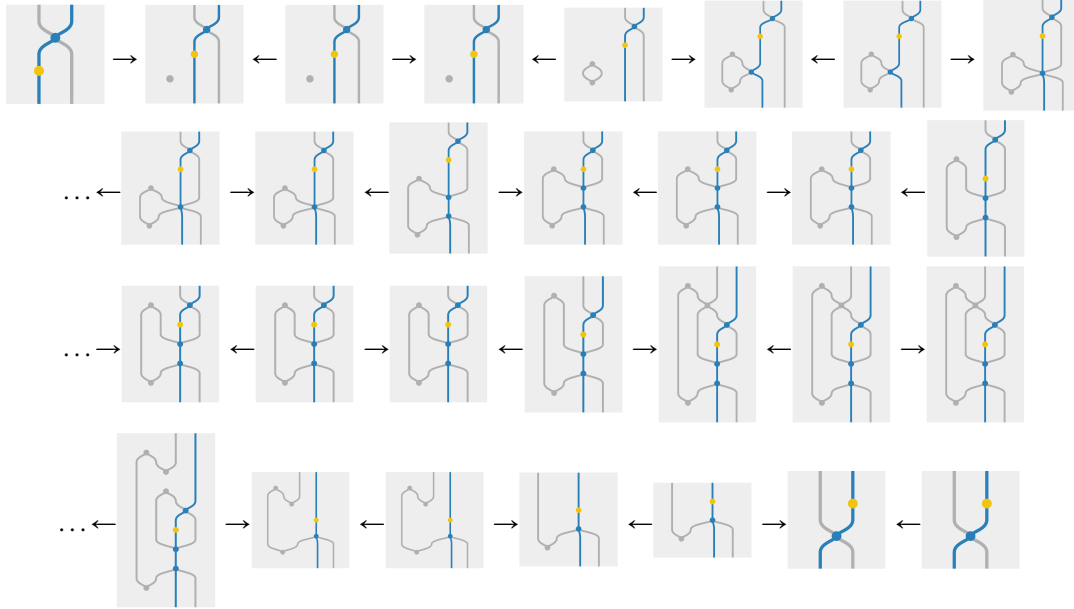Fig. 10: Constructing the 4-dimensional naturality homotopy.

Fig. 11: Constructing the 5-dimensional naturality of naturality homotopy.

[3] T. Leinster, *Higher Operads, Higher Categories*. CUP, 2003, arXiv:math/0305049.

[4] R. Gordon, J. Power, and R. Street, "Coherence for tricategories," *MAMS*, vol. 117.

[5] P. Selinger, "A survey of graphical languages for monoidal categories," in *New Structures for Physics*, ser. Lecture Notes in Physics, vol. 813, 2011, pp. 289–233.

[6] J. Baez and J. Dolan, "Higher-dimensional algebra and topological quantum field theory," *JMP*, vol. 36, pp. 6073–6105, 1995, arXiv:q-alg/9503002.

[7] "The proof assistant *homotopy.io*," *n*Lab page.

[8] K. Bar and J. Vicary, "Data structures for quasistrict higher categories," in *Proceedings of LICS 2017*, arXiv:1610.06908.

[9] K. Bar, A. Kissinger, and J. Vicary, "Globular: an online proof assistant for higher-dimensional rewriting," *LMCS*, vol. 14, 2018.

[10] A. Delpeuch and J. Vicary, "Normal forms for planar string diagrams," 2018, arXiv:1804.07832.

[11] M. Makkai, "The word problem for computads," 2005, Available online.

[12] J. Baez, A. Hoffnung, and C. Walker, "Higher-dimensional algebra vii: Groupoidification," 2009, arXiv:0908.4305.

[13] M. Grandis, "Cubical cospans and higher cobordisms," *JHRS*, 2008, to appear. arXiv:0806.2359.

[14] J. Morton, "Double bicategories and double cospans," *JHRS*, vol. 4, pp. 389–428, 2009, arXiv:math/0611930.

[15] M. Stay, "Compact closed bicategories," 2013, arXiv:1301.1053.

[16] A. Joyal and R. Street, "Braided tensor categories," *Advances in Mathematics*, vol. 102, no. 1, pp. 20–78, 1993.