

# Data structures for quasistrict higher categories

Krzysztof Bar

Department of Computer Science  
University of Oxford, United Kingdom  
krzysztof.bar@cs.ox.ac.uk

Jamie Vicary

Department of Computer Science  
University of Oxford, United Kingdom  
jamie.vicary@cs.ox.ac.uk

**Abstract**—We present new data structures for quasistrict higher categories, in which associativity and unit laws hold strictly. Our approach has low axiomatic complexity compared to traditional algebraic definitions of higher categories, and we use it to give a practical definition of quasistrict 4-category. It is amenable to computer implementation, and we exploit this to give a machine-verified algebraic proof that every adjunction of 1-cells in a quasistrict 4-category can be promoted to a coherent adjunction satisfying the butterfly equations.

## I. INTRODUCTION

### A. Our contribution

**Motivation.** Higher category theory now plays an essential role in many areas of mathematics, physics and computer science. Among the most striking applications are the homotopy type theory programme on univalent foundations for mathematics [1–3], motivated by the work of Hofmann and Streicher on an intensional groupoid model for Martin-Löf type theory [4]; the outline proof by Lurie [5] of the cobordism hypothesis of Baez and Dolan [6], and the associated revolution in topological quantum field theory [7, 8]; and Lurie’s higher topos theory programme [9], developing ideas going back to Grothendieck [10], with broad implications for geometry. Other applications in computer science include concurrency [11, 12], rewriting [13–15], and quantum computation [16, 17].

Nonetheless, higher category theory is “generally regarded as a technical and forbidding subject” [9]. This may be in part because of the complexity of the definitions from an algebraic perspective. For example, even in the semistrict case where as much structure as possible is suppressed, semistrict 2-categories comprise 3 sets equipped with 6 functions satisfying 12 equations [18]; semistrict 3-categories comprise 4 sets equipped with 19 functions satisfying 58 equations [19]; and semistrict 4-categories comprise 5 sets equipped with 34 functions satisfying 118 equations [20]<sup>1</sup>. This complexity masks to some extent the naturalness of these structures, and can make them hard to work with directly when constructing or verifying proofs. Traditional proof assistants such as Coq cannot easily help with these difficulties<sup>2</sup>, not least because

they do not support the pasting diagram or string diagram notations which are prevalent in higher category theory.

In this paper we present a new approach to defining and working with globular<sup>3</sup> higher categories, applying in the *quasistrict* case, meaning that composition is strictly associative and unital.<sup>4</sup> Our approach allows us to give a concise definition of semistrict 4-category, which corrects some previous errors in the literature. We also give a definition of quasistrict 4-category, a weaker structure which is more practical for the purpose of constructing proofs. Our proposal is computationally implemented, and we give details of a substantial formalized proof developed in the quasistrict 4-category setting, giving evidence of the correctness and practicality of our approach.

**Signatures and diagrams.** A group can be defined explicitly in terms of a set of elements, or implicitly in terms of a *presentation* involving generators and relations. Similarly, we work with higher categories in terms of a presentation, encoded by a *signature* that gives the generating objects, 1-cells, 2-cells, and so on, up to  $n$ -cells for some  $n \in \mathbb{N}$ . For  $k > 0$ , the generating  $k$ -cells are equipped with source and target  $(k-1)$ -diagrams, encoded in terms of *diagram* structures, a new concept that we introduce.

A diagram of dimension  $k > 0$  comprises a source  $(k-1)$ -diagram, a list of generating  $k$ -cells, and a list of attachment coordinates for these generators. It describes a construction procedure: to build the composite  $k$ -diagram, begin with the source  $(k-1)$ -diagram, and attach the generating  $k$ -cells sequentially at their specified attachment coordinates. As a result, each  $k$ -cell in the diagram is at a unique *height*, given by its position in the list, and our diagrams satisfy a *generic position* property similar of the central property in higher Morse theory [8].

A key strength of our approach is reduced axiomatic complexity, and we can explain the source of this reduction informally. Let  $D$  be a traditional algebraic definition of some flavour of higher category, which is at least strictly associative and unital, such as one of the semistrict definitions mentioned above. Some of the function data in  $D$  will correspond to vertical composition operations, which specify how two  $k$ -cells compose to produce another  $k$ -cell; in our approach, vertical

<sup>1</sup>These counts are conservative estimates obtained from close readings of the definitions, and are to some degree subjective. There are some issues with this definition of semistrict 4-category (see Section I-B), but these do not affect the magnitudes of these numbers. We neglect set-theoretic issues.

<sup>2</sup>Homotopy type theory is perhaps an exception; see Section I-B.

<sup>3</sup>This is a standard approach to higher category theory [21], in which for a  $k$ -morphism  $f$  for  $k \geq 2$  we have  $s(s(f)) = s(t(f))$  and  $t(s(f)) = t(t(f))$ , where  $s$  and  $t$  give the source and target respectively.

<sup>4</sup>This includes the semistrict case mentioned above.

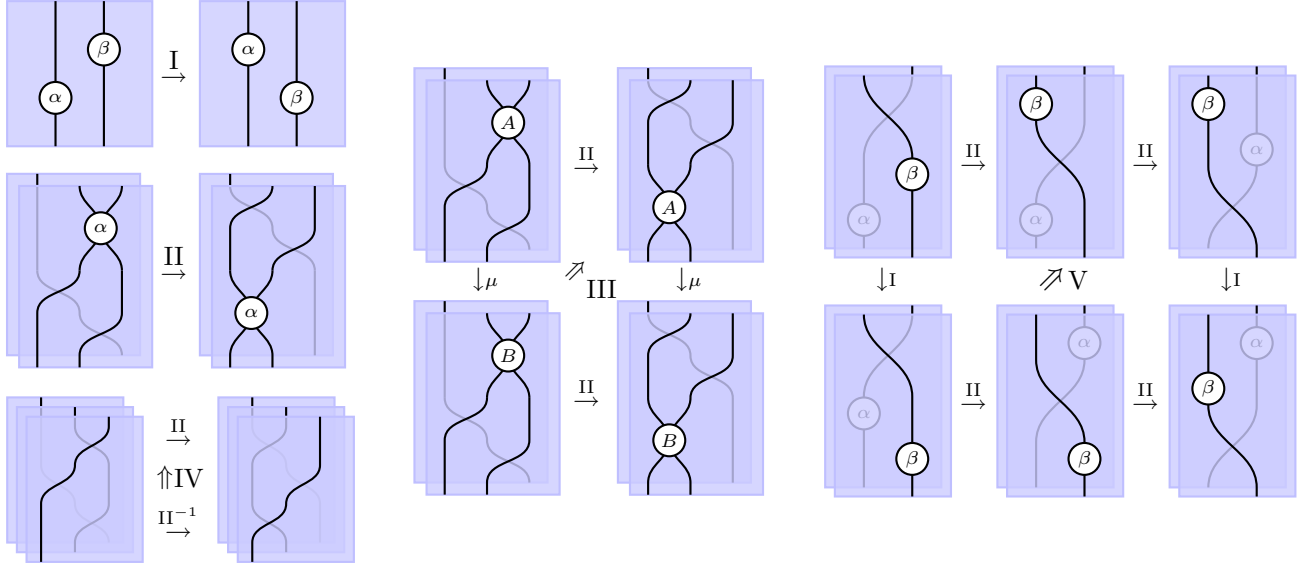


Fig. 1: The five homotopy generator families.

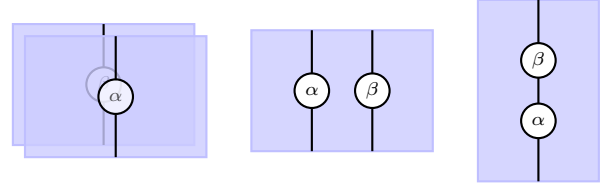
composition is simply concatenation of the appropriate lists, and no additional composition data is needed. Furthermore, some of the equation data in  $D$  will encode strict associativity or unitality of vertical composition; in our approach these equations can be neglected, since it is a trivial property of lists that concatenation is strictly associative and unital, with the unit given by the empty list. Indeed, these equations become *theorems* in our approach, and the major technical work of this paper is proving theorems of this kind.

For our new definitions, outlined below, we obtain the following: a semistrict 2-category comprises 4 sets equipped with 7 functions satisfying 4 equations; a semistrict 3-category comprises 5 sets equipped with 11 functions satisfying 6 equations; and a semistrict 4-category comprises 6 sets equipped with 18 functions satisfying 8 equations<sup>5</sup>. Note that we must consider more sets than before, since, for example, a presentation of a 2-category comprises not only generating objects, 1-cells and 2-cells, but also generating equations. However, these sets can be finite even in nontrivial cases, since infinite categories can have finite presentations<sup>6</sup>; this is the source of another substantial reduction in practice of complexity of the required data.

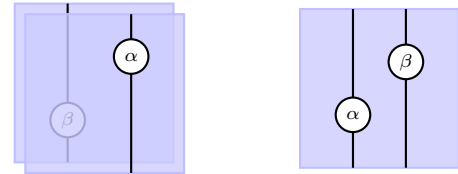
**Graphical calculus.** Based on ideas of Trimble [22], we sketch in Section IV an informal graphical calculus for  $n$ -diagrams, which can be made precise for dimensions  $n \leq 3$ , in which an  $n$ -diagram is represented as a labelled partitioned subspace of  $\mathbb{R}^n$ . This is consistent with previous proposals in dimension  $n \leq 3$  [23–26]. However, a unique feature of our diagrams is their similarity to generic-position Morse

diagrams [8], a crucial feature of our approach which we now briefly explore.

In ordinary algebraic approaches to higher category theory, a  $p$ -cell and a  $q$ -cell can be composed in  $\min(p, q)$  ways. Two 3-cells  $\alpha, \beta$  can therefore be composed in 3 ways, which we can illustrate graphically as follows:



In the first diagram the two 3-cells are overlapping, while in the second they are at the same height; only the third diagram is in generic position, which is an essential requirement of our representation, as discussed above. This is a general phenomenon: of all the  $\min(p, q)$  ways to compose a  $p$ -cell and a  $q$ -cell, only the highest-dimensional composition yields a generic position diagram. We internalize this in our formalism, allowing *at most one* composition of  $p$ -diagram and a  $q$ -diagram, along a common boundary  $(\min(p, q) - 1)$ -cell; this smaller number of composition operations helps to further reduce the complexity of our algebraic system. Generic-position perturbations of the missing composites can still be accessed by repeated whiskering and composition, as follows:



In this way we retain full compositional expressivity while reaping the advantages of a generic-position representation.

<sup>5</sup>As before, these counts are to some extent subjective estimates; the point is that they are an order of magnitude smaller than for conventional definitions.

<sup>6</sup>That this happens for some higher categories of manifolds follows from the cobordism hypothesis, and indeed is one of the main reasons for interest in the hypothesis.



Fig. 2: Different parameterizations of the same homotopy in a Gray category.

**Homotopy generators.** Diagram and signature structures give efficient tools for handling higher-dimensional composition, but lack a notion of *homotopy* which is fundamental to higher category theory. In Section V we supply this in terms of *homotopy generators* of certain types, previewed in simple form in Figure 1. These must be described separately in each dimension. We use them to give new definitions of semistrict 2-, 3- and 4-category as follows.

**Definition 1.** A *semistrict 2-category* is a 3-signature supporting homotopy generators of type I.

**Definition 2.** A *semistrict 3-category* is a 4-signature supporting homotopy generators of types I and II.

**Definition 3.** A *semistrict 4-category* is a 5-signature supporting homotopy generators of types I, II, III, IV and V.

These are simpler than traditional algebraic definitions. This simplicity arises in part from the nature of signature and diagram structures, as discussed above. However, our treatment of homotopy generators yields two further simplifications.

Firstly, we are able to neglect equations governing redundant encodings of the same homotopy. For example, Figure 2 illustrates two equivalent descriptions of the same 3-cell interchanger in a Gray category, based on different implicit descriptions of the source diagram; Gray categories require an axiom that says  $\Phi_1 = \Phi_2$ . In our approach, this homotopy arises in a unique way, with no redundancy in its description, and therefore with no need for additional equations to control this redundancy.

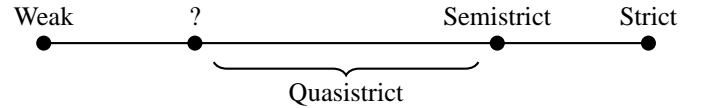
Secondly, our approach is able to treat simultaneously homotopy generators of the same kind appearing in different dimensions, while traditional algebraic definitions must treat them separately. For example, in a Gray category, there are interchangers of 2-cells that yield 3-cells such as  $\Phi_1$ ,  $\Phi_2$  of Figure 2, as well as interchangers of 3-cells that yield equations, axiomatized separately. We treat these uniformly in Definition 2, as they all arise as instances of the type I homotopy generator.

The relevance of these homotopy generator classes to higher category theory is already well-known; for example, they all appear in some form in the definition of braided monoidal 2-category due to Baez and Neuchl [27]. The novelty here is the way they are treated, and the relative axiomatic simplicity we obtain, as emphasized above.

**Quasistrict  $n$ -categories.** A definition of  $n$ -category is semistrict when it is ‘as strict as possible’, while still being able to model arbitrary homotopy  $n$ -types. When studying

$n$ -categories it can be attractive to have a semistrict definition available, to get a sense of the minimal algebraic complexity of the theory. However, if our desire is to prove theorems internal to our  $n$ -categories, then there is a more important concern: shorter proofs. A simpler definition of  $n$ -category can sometimes lead to shorter proofs, but if taken too far it can have the opposite effect: the language might become so meagre that while everything remains possible in principle, some conceptually-simple proofs become long-winded.

We illustrate these ideas with the following heuristic diagram of the weak-strict spectrum:<sup>7</sup>



Here, ‘weak’ marks the weakest possible definition of  $n$ -categories, such as that arising from the contractible operads described by Batanin and Leinster [21, 28]; ‘strict’ marks strict  $n$ -categories. We now define the term ‘quasistrict’.<sup>8</sup>

**Definition 4.** A definition of  $n$ -categories is *quasistrict* if it is strictly associative and unital, and can model all homotopy  $n$ -types.

Quasistrict  $n$ -categories therefore occupy a region in the centre of the weak-strict spectrum. Semistrict  $n$ -categories are the strictest quasistrict  $n$ -categories<sup>9</sup>. We propose that the definition of  $n$ -categories most amenable to computation will be the *weakest* quasistrict definition, in some sense; this is marked ‘?’ on the diagram above, since we do not know how to give such a definition.

Here we give a definition of quasistrict 4-category which is somewhere in the interior of the quasistrict region of the weak-strict spectrum.

**Definition 5.** A *quasistrict 4-category* is a 5-signature supporting homotopy generators of types I, II’, III’, IV’, V’ and VI.

We slightly abuse terminology here; this is merely *a* definition of quasistrict 4-category, not *the* definition, since ‘quasistrict’ is a broad notion. Some of these homotopy generator classes are stronger than those used in the definition of semistrict

<sup>7</sup>We do not mean to imply with this informal diagram that definitions of  $n$ -category form a total order in any precise sense.

<sup>8</sup>This definition is consistent with previous uses of the term by Bartlett [29] and Schommer-Pries [8].

<sup>9</sup>Here we rule out some approaches to semistrict higher categories, such as Simpson’s snucategories [30] with strict interchangers and weak units. Such approaches are of course entirely valid, but inconsistent with our methods.

4-category given above, allowing some deductions which would be long-winded in the semistrict case to be given in 1 step in the quasistrict case.

**Computer formalization.** Since quasistrict 4-categories are designed to aid proof construction, it makes sense to formalize them using in a proof assistant. Based on the ideas of this paper, and in collaboration with Kissinger, the authors have developed *Globular* [31]<sup>10</sup>, a proof assistant which allows the user to construct formal proofs in finitely-presented quasistrict 4-categories in the sense of Definition 5. The interface is fully graphical, allowing construction and interaction with proof objects by clicking-and-dragging, and the tool is available online through a web page, minimizing barriers to use and allowing formal proofs to be hyperlinked directly from research papers. The proof assistant has been well-received by the community, being accessed 10,304 times by 2,530 distinct users in the first 13 months since launch in December 2015.<sup>11</sup> The system allows users to make their proofs visible publicly, and to date 49 proofs have been published using this mechanism, on topics including higher category theory and coherence, classical knots, knotted embedded surfaces, sphere eversion, Hopf algebras, automata and linear logic.

In particular, the current authors have used this proof assistant to develop a new result, which is the final main result presented in this article: in a quasistrict 4-category, an adjunction of 1-morphisms can be promoted to give a coherent adjunction satisfying the butterfly equations. This proof is available online at [globular.science/1605.002](http://globular.science/1605.002) for direct inspection, and presented in detail in Appendix A.

By general results of Riehl and Verity [32], it is expected that such a theorem will hold in any reasonable algebraic definition of 4-category. Our proof is the first such that has been given; indeed, we believe it to be the first nontrivial proof of any sort internal to an algebraic 4-category, that has been presented in the literature.<sup>12</sup> That we were able to construct this proof is the strongest evidence we can supply for the correctness and utility of our framework, and for the definition of quasistrict 4-category that we have built within it.

**Criticisms.** We criticise our work in the following ways. Firstly, we do not prove that our definitions of semistrict 2-, 3- and 4-categories are equivalent to the standard weak notions [33, 34]. In the case of 2- and 3-categories we are very confident that this is the case, and in the case of 4-categories we are reasonably confident; as evidence, we show in [35, Section 3.8] that a Gray category gives rise to a semistrict 3-category in our sense, and in [35, Section 3.7] that a 4-tas in the sense of Crans [20] gives rise to a semistrict 4-category (modulo some issues that we identify with Crans’ definition.) It is worth noting that for traditional semistrict

3-categories, it took 20 years for such an equivalence theorem to be proved, between Gray’s work [36, 37] around 1974 and the coherence theorem of Gordon, Power and Street [33] in 1995; furthermore, the definition of semistrict 4-category due to Crans has never been shown equivalent to the tetracategories of Trimble [34, 38], or to any other definition of 4-category.

Secondly, while we hold that our methods are simpler than existing approaches, we do not propose a definition of semistrict 5-category. The basic foundation of signature and diagram structures developed here should continue to be usable in any finite dimension, and indeed the many properties proved in Sections II and III are dimension-agnostic. We expect that semistrict 5-categories would require a total of 9 homotopy generator families; identifying these would require careful manual analysis. It would be better to find a more systematic approach to specifying the homotopy generators automatically, preferably in a way which moves closer to the weak end of the weak-strict spectrum as discussed above.

**Acknowledgements.** We are extremely grateful to Aleks Kissinger for many detailed discussions, especially in the early stages of this work. We also thank John Baez, Manuel Bärenz, Bruce Bartlett, Eugenia Cheng, Chris Douglas, Eric Finster, Nick Gurski, André Henriques, Samuel Mimram, Christopher Schommer-Pries, Dominic Verdon and Steve Vickers for useful conversations, and the audiences of HDRA 2015, HDRA 2016 and CT 2016 for feedback on early results.

**Proofs.** Proofs are omitted in this extended abstract due to their significant length, and can be found in the full paper [35].

## B. Related work

**Crans semistrict 4-categories.** Crans has given a definition of semistrict 4-category [20], under the name ‘4-tas’. Our definition, while presented using our new approach, is broadly consistent. However, the definitions are certainly inequivalent, in particular due to the absence of certain structures in a 4-tas:

- there is no equivalent to our type IV homotopy generators;
  - there is no equivalent to our type I homotopy generator rearrangement, written  $\hat{I}$  in our formalism, which play a key role in the description of the type V homotopy generators.
- In the full article [35] we show that aside from these issues, every 4-tas yields a semistrict 4-category in our sense.

**Homotopy type theory (HoTT).** This broad research programme seeks to develop a new foundation for mathematics, in a setting based on a homotopical interpretation of Martin-Löf’s dependent type theory [1]. It is amenable to computer formalization, and there has been substantial activity in producing formal proofs [39, 40].

While the motivations are related, there is little overlap between proofs suitable for formalization in HoTT and in our system. HoTT is far more expressive, with a rich syntax of term constructors that go well beyond the basic higher composition operations considered here. At the same time, the direct access to homotopy generators that our system provides allows direct construction of sophisticated homotopies—such as the complex proof illustrated in Appendix A, or many of

<sup>10</sup>This is a tool paper which focuses on the user interface and implementation, giving only an outline sketch of the theoretical foundations. The present work is the corresponding theory paper.

<sup>11</sup>Data from Google Web Analytics retrieved on 7 January 2017.

<sup>12</sup>We distinguish here between algebraic definitions of 4-category (such as those due to Trimble and Crans, and that described here), and homotopical ones (such as quasicategories) which build in the homotopy hypothesis.



the examples listed at [ncatlab.org/nlab/show/Globular](http://ncatlab.org/nlab/show/Globular)—which would be hard to construct directly in a formal system for HoTT. Furthermore, our system does not require invertibility of higher cells, while the natural categorical interpretation of homotopy type theory is in an  $(\infty, 1)$ -setting where  $k$ -cells for  $k > 1$  are invertible.

**Rewriting theory.** There is a large body of work on polygraphs and their applications to higher-dimensional rewriting [13–15, 41]. The setting is strict  $n$ -categories, and polygraphs directly correspond to our signature structures. Our work owes a lot to the perspective developed by this community, which emphasizes the idea that a composite  $(n+1)$ -cell is precisely the same as a rewrite sequence of  $n$ -cells. Our research programme can be considered an attempt to apply this perspective in the more general semistrict setting. Central questions of study in the polygraph community are confluence and normalization, which we neglect here. The basic techniques are also different: while the polygraph community relies on a pushout formalism for constructing rewrites [42], our approach is more combinatorial. Furthermore, our diagram structures do not apply in the strict setting, since they contain layout data which breaks the strict polygraph setting.

**Enriched approaches.** It is natural to propose defining semistrict  $(n+1)$ -categories recursively, as categories enriched in the category of semistrict  $n$ -categories. This heuristic idea is successful in leading to a definition of semistrict 3-category, but fails to give a definition of semistrict 4-category because the relevant tensor product of Gray categories is not monoidal closed, an argument sketched by Crans [43] and refined by Bourke and Gurski [44]. While this issue remains unresolved, work by Batanin, Cisinski, Garner and Weber [45–49] suggests an operadic perspective allowing recursive enriched definitions of higher categories with strict units, which would include the semistrict case. However, this approach has not yet led to a concrete definition of semistrict 4-category, and it is unclear whether our definition of semistrict 4-category could arise in principle from their work.

**Opetopic higher categories.** Baez and Dolan’s opetopic theory of higher categories [50], given a combinatorial interpretation in terms of trees by Kock and collaborators [51], has been developed by Finster [52] as the foundation for a higher-dimensional type theory. This has been implemented as the proof assistant *Opetopic!* [53], allowing description of formal proofs in opetopic  $(\infty, \infty)$ -categories. The overall structure is at once elegant and powerful, with the tree interpretation giving a graphical calculus, albeit one which is quite different from the generic-position diagrams used here. Opetopic categories have a more restricted form of pasting diagram than globular categories, however, and the approaches are not in general expected to be equivalent. Furthermore, we are not aware of a substantial formalized proof in the opetopic setting, comparable to the proof we present in Section A.

## II. DATA STRUCTURES

Here we introduce the basic data structures that underlie our approach: *signatures*, *diagrams*, *embeddings*, *slices*, *rewrites* and *lifts*. These are mutually recursive, so we will necessarily refer to some of them before they are formally defined. We therefore begin with informal definitions, as follows:

- an  $n$ -signature  $\Sigma$  is the data of a presentation in dimension  $n$ , specifying the available generating  $k$ -cells  $g$  in each dimension  $k \leq n$ , each having (for  $k > 0$ ) source and target  $(k-1)$ -diagrams  $g.s$  and  $g.t$  respectively;
- an  $k$ -diagram  $D$  over an  $n$ -signature  $\Sigma$  (for  $k \leq n$ ) is a composite of generating  $k$ -cells, with (for  $k > 0$ ) source and target  $(k-1)$ -diagrams  $D.s$  and  $D.t$ ;
- an *embedding* is the data witnessing one diagram as a subdiagram of another;
- a *slice* of an  $n$ -diagram  $D$  is an  $(n-1)$ -diagram defined by  $D$  at some constant height;
- a *rewrite* is the diagram  $D'$  obtained by removing a subdiagram  $S_1$  of a given diagram  $D$ , and replacing it with another diagram  $S_2$ ;
- a *lift* is the canonical embedding of  $S_2$  within  $D'$  induced by such a rewrite.

The key structure is that of signature: in Section I-A we define semistrict and quasistrict  $n$ -categories for  $n \leq 4$  as  $(n+1)$ -signatures with particular properties. Signatures are similar to polygraphs as used in the rewriting community (see Section I-B); however, our notion of diagram is unrelated, and can be considered the central innovation. We give conditions for instances of these structures to be well-defined, and give many propositions establishing that slices, rewrites and lifts of well-defined diagrams are again well-defined.

### A. Signatures and diagrams

We begin with the definitions of signature and diagram.

**Definition 6** (Signature). For  $n \geq 0$ , an  $n$ -signature  $\Sigma$  is a family of sets  $(\Sigma_0, \dots, \Sigma_n)$ , such that when  $n > 0$ :

- $(\Sigma_0, \dots, \Sigma_{n-1})$  is an  $(n-1)$ -signature  $\Sigma'$ ;
- each  $g \in \Sigma_n$  is equipped with  $(n-1)$ -diagrams  $g.s, g.t$  over  $\Sigma'$ , such that when  $n > 1$ ,  $g.s.s = g.s.t$  and  $g.t.s = g.t.t$ .

An  $n$ -signature gives rise to a canonical  $k$ -signature for any  $k < n$  in an obvious way, and we often use this implicitly.

**Definition 7** (Diagram). For  $n \geq 0$ , an  $n$ -diagram  $D$  over an  $n$ -signature  $\Sigma$  is a list of length  $|D|$ , with each element  $D[i]$  for  $0 \leq i < |D|$  comprising the following:

- a generating  $n$ -cell  $D[i].g \in \Sigma_n$ ;
- if  $n > 0$ , an embedding  $D[i].e : D[i].g.s \hookrightarrow D[i].d$ ;
- if  $n > 0$ , a source  $(n-1)$ -diagram  $D.s$  over  $\Sigma$ .

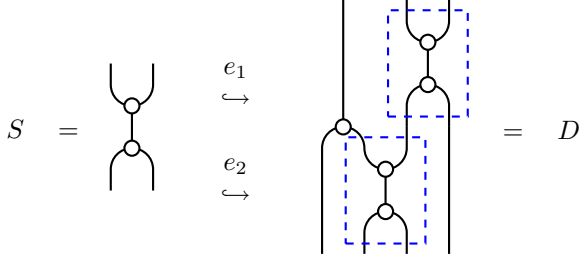
Furthermore, if  $n = 0$  then we must have  $|D| = 1$ .

We now consider what it means for a diagram to be well-defined. This makes use of the notion of *slice*, defined formally later in this section. Informally, the slices of an  $n$ -diagram  $D$  for  $n > 0$  are  $(n-1)$ -diagrams  $D[i].d$  appearing at intermediate heights  $0 \leq i \leq |D|$  within the diagram. The zeroth slice  $D[0].d$  equals the source  $D.s$ .

**Definition 8.** An  $n$ -diagram  $D$  over an  $n$ -signature  $\Sigma$  is *well-defined* when  $n = 0$ , or when  $n > 0$  and the source diagram  $D.s$  is well-defined, and for every  $0 < i \leq |D|$  the slice  $D[i].d$  exists and is well-defined.

### B. Embeddings

For  $n$ -diagrams  $S, T$ , on an intuitive level, an embedding  $e : S \hookrightarrow D$  specifies a way in which  $S$  appears as a subdiagram of  $D$ . We illustrate this with the following example of a pair of 2-diagrams, with two embeddings  $e_1, e_2 : S \hookrightarrow D$ :



The formal definition goes as follows.

**Definition 9** (Embedding). Given two  $n$ -diagrams  $S$  and  $D$ , an *embedding*  $e : S \hookrightarrow D$  consists of no data when  $n = 0$ , and of the following data when  $n > 0$ :

- an embedding height  $e.h \in \mathbb{N}$ ;
- a source embedding  $e.e : S.s \hookrightarrow D[e.h].d$ .

Embeddings are therefore specified by sequences of natural numbers. The first embedding above has values  $e_1.h = 0$  and  $e_1.e.h = 1$ , meaning that its image begins at height 0 and has 1 wire to the left. The second embedding  $e_2$  has values  $e_2.h = 3$  and  $e_2.e.h = 1$ , meaning that its image begins at height 3 and has 1 wire to the left. Note that there are no vertices to the left or right of the images of  $e_1$  and  $e_2$ , and so in particular the number of wires to the left and right of each image is determinate; this follows from the well-definedness property given below.

Just as for diagram structures, we give a formal statement of what it means for an embedding to be well-defined.

**Definition 10.** An  $n$ -diagram embedding  $e : S \hookrightarrow D$  between well-defined diagrams is *well-defined* if it satisfies the following properties. If  $n = 0$ , we have

$$\bullet S[0].g = D[0].g; \quad (1)$$

otherwise if  $n > 0$ , we have:

$$\bullet \text{ the component embedding } e.e \text{ is well-defined;} \quad (2)$$

$$\bullet S[i].g = D[i + e.h].g \quad (2)$$

$$\bullet (e.e.\Lambda[S[i].d]) \circ S[i].e = D[i + e.h].e \quad (3)$$

The symbol  $\Lambda$  represents the lift construction, described below. Intuitively, condition (2) says that the generators of  $S$  are the same of those of  $D$  in the image of the embedding, and condition (3) says that the embedding maps are compatible.

### C. Rewriting

Rewriting transforms a diagram  $D$ , by removing a subdiagram  $e : S \hookrightarrow D$ , and replacing it with another diagram  $T$ , which is then a subdiagram of the rewritten diagram  $D.\Pi[e, T]$  (see Figure 3.) In order for  $T$  to ‘fit in the hole’ left after removing  $S$ , we need  $S.s = T.s$  and  $S.t = T.t$ , and we need to update the relevant embeddings  $T[i].e$  which tell us how the generators of  $T$  sit inside  $D.\Pi[e, T]$ . The generators of  $D.\Pi[e, T]$  come in 3 families: those arising from  $D$  below the image of  $S$ ; those arising from  $T$ ; and those arising from  $D$  above the image of  $S$ .

**Definition 11.** Two  $k$ -diagrams  $S, T$  over the same signature are *globular* when  $k = 0$ , or when  $k > 0$  and  $S.s = T.s$  and  $S.t = T.t$ .

**Definition 12** (Rewrite). Given an  $n$ -diagram  $D$  with an embedding  $e : S \hookrightarrow D$  such that  $S, T$  are globular, the *rewrite*  $D.\Pi[e, T]$  is the  $n$ -diagram such that

$$\bullet |D.\Pi[e, T]| = |D| - |S| + |T|; \quad (4)$$

and such that if  $n = 0$ , then

$$\bullet D.\Pi[e, T][0].g = T[0].g; \quad (5)$$

and such that if  $n > 0$ , then

$$\bullet D.\Pi[e, T].s = D.s; \quad (6)$$

$$\bullet D.\Pi[e, T][i].g = \begin{cases} D[i].g & 0 \leq i \leq e.h \\ T[i - e.h].g & e.h < i < e.h + |T| \\ D[i - |T| + |S|].g & e.h + |T| \leq i < |D| - |S| + |T|; \end{cases} \quad (7)$$

$$\bullet D.\Pi[e, T][i].e = \begin{cases} D[i].e & 0 \leq i \leq e.h \\ e.e.\Lambda[T[i - e.h].d] \circ T[i - e.h].e & e.h < i < e.h + |T| \\ D[i - |T| + |S|].e & e.h + |T| \leq i < |D| - |S| + |T|. \end{cases} \quad (8)$$

The definition of rewrite gives rise to the definition of slice.

**Definition 13** (Slice). Given an  $n$ -diagram  $D$ , for  $0 < i \leq |D|$ , its *slice*  $D[i].d$  is the following  $(n-1)$ -diagram:

- if  $i = 0$ ,  $D[0].d := D.s$ ;
- if  $i > 0$ ,  $D[i].d := D[i-1].d.\Pi[D[i-1].e, D[i-1].g.t]$

We use the concept of slice to define the target of a diagram, and the notion of globularity of a pair of diagrams.

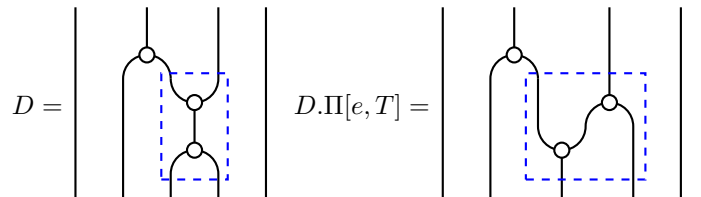


Fig. 3: Rewriting.

**Definition 14.** For  $n > 0$ , given an  $n$ -diagram  $D$ , its *target* is the  $(n-1)$ -diagram  $D.t := D[|D|].d$ .

#### D. Lifts

Given an embedding  $e : S \hookrightarrow D$ , then  $T$  embeds into  $D.\Pi[e, T]$  in a natural way. We call this a lifted embedding.

**Definition 15.** For  $n \geq 0$ , given globular  $n$ -diagrams  $S, T$  and an embedding  $e : S \hookrightarrow D$ , the *lifted embedding*  $e.\Lambda[T] : T \hookrightarrow D.\Pi[e, T]$  is defined as follows:

- $e.\Lambda[T].h = e.h$
- $e.\Lambda[T].e = e.e$

Lifted embeddings are used to construct the embeddings in Definition 12 of a rewrite.

We also use the lifted embedding to define composition of two embeddings  $e : S \hookrightarrow D$  and  $f : D \hookrightarrow A$ , given below. This is necessary since there is a mismatch between the source of  $f.e$  and the target of  $e.e$ , and  $f.e.\Lambda[D[e.h].d]$  is needed to make the transition between them.

**Definition 16.** For  $n \geq 0$ , given  $n$ -diagram embeddings  $e : S \hookrightarrow D$  and  $f : D \hookrightarrow A$ , their *composite embedding*  $f \circ e : S \hookrightarrow A$  is defined as follows for  $n > 0$ :

- $(f \circ e).h := e.h + f.h$
- $(f \circ e).e := (f.e.\Lambda[D[e.h].d]) \circ e.e$

For  $n = 0$ , it is defined to have no data.

This has a clear interpretation: that the notion of subdiagram is transitive.

#### E. Equivalence

We now introduce notions of equivalence for diagrams and embeddings. These notions are mutually recursive.

**Definition 17.** Two  $n$ -diagram embeddings  $e : A \hookrightarrow B$  and  $f : C \hookrightarrow D$  are *equivalent*, written  $e = f$ , when  $A = C$  and  $B = D$ , and furthermore if  $n > 0$  when  $e.h = f.h$  and  $e.e = f.e$ .

**Definition 18.** Two  $n$ -diagrams  $D$  and  $S$  are *equivalent*, written  $D = S$ , when:

- $|S| = |D|$ ;
- for  $0 \leq i < |D|$  we have  $S[i].g = D[i].g$ ;
- if  $n > 0$ , for  $0 \leq i < |D|$  we have  $S[i].e = D[i].e$ ;
- if  $n > 0$ , then  $D.s = S.s$

#### F. Well-definedness

Our first major family of results establishes well-definedness of our constructions.

**Well-definedness.** When our basic procedures operate on well-defined structures, the result is again well-defined.

**Proposition 19** (Well-defined rewrites). *For  $n \geq 0$ , given well-defined  $n$ -diagrams  $D, S, T$  with  $S.s = T.s$  and  $S.t = T.t$ , and a well-defined embedding  $e : S \hookrightarrow D$ , the rewrite  $D.\Pi[e, T]$  is well-defined.*

**Proposition 20** (Well-defined lifts). *For  $n \geq 0$ , given well-defined  $n$ -diagrams  $S, T, A$  with  $S.s = T.s$  and  $T.s = T.t$ ,*

*and given a well-defined embedding  $e : S \hookrightarrow A$ , then the lifted embedding  $e.\Lambda[T] : T \hookrightarrow A.\Pi[e, T]$  is well-defined.*

**Proposition 21** (Well-defined composition). *For  $n, m \geq 0$ , given an  $n$ -diagram  $D$  and an  $m$ -diagram  $S$ , both well-defined, such that either  $n \geq m$  and  $S.t = s^{n-m+1}(D)$ , or  $m > n$  and  $t^{m-n+1}(S) = D.s$ , then  $S \circ D$  is well-defined.*

**Proposition 22** (Well-defined composite embeddings). *For  $n \geq 0$ , given well-defined  $n$ -diagram embeddings  $e : S \hookrightarrow D$  and  $f : D \hookrightarrow M$ , then  $f \circ e : S \hookrightarrow M$  is well-defined.*

As an example of our methods we give here a full proof for a simple lemma.

**Lemma 23.** *Given an  $n$ -diagram  $D$  and  $e : S \hookrightarrow D$ , the following equivalence of diagrams holds:*

$$D.\Pi[e, S] = D$$

*Proof.* By Definition 18, we need to check four conditions.

**Sources are equivalent diagrams.**

$$D.\Pi[e, S].s = D.s \quad [\text{Eq. (6)}]$$

**Sizes of lists of generators are equal.**

$$|D.\Pi[e, S]| = |D| - |S| + |S| = |D| \quad [\text{Eq. (4)}]$$

**Corresponding generators are equal.** We must consider this for  $0 \leq j \leq |D|$ . We consider this separately for three ranges. For  $0 \leq j \leq e.h$ :

$$D.\Pi[e, S][j].g = D[j].g \quad [\text{Eq. (7)}]$$

For  $e.h \leq j \leq e.h + |S|$ :

$$D.\Pi[e, S][j].g = S[j - e.h].g \quad [\text{Eq. (7)}]$$

$$= D[(j - e.h) + e.h].g \quad [\text{Eq. (3)}]$$

$$= D[j].g$$

For  $e.h + |S| \leq j \leq |D|$ :

$$D.\Pi[e, S][j].g = D[j - |S| + |S|].g \quad [\text{Eq. (7)}]$$

$$= D[j].g$$

**Corresponding embeddings are equivalent.** We must consider this for  $0 \leq j \leq |D|$ . We consider this separately for three ranges. For  $0 \leq j \leq e.h$ :

$$D.\Pi[e, S][j].e = D[j].e \quad [\text{Eq. (8)}]$$

For  $e.h \leq j \leq e.h + |S|$ :

$$D.\Pi[e, S][j].e = (e.e.\Lambda[S[j - e.h].d]) \circ S[j - e.h].e \quad [\text{Eq. (8)}]$$

$$= D[(j - e.h) + e.h].e \quad [\text{Eq. (3)}]$$

$$= D[j].e$$

For  $e.h + |S| \leq j \leq |D|$ :

$$D.\Pi[e, S][j].e = D[j - |S| + |S|].e = D[j].e \quad [\text{Eq. (8)}]$$

This completes the proof.  $\square$

### III. COMPOSITION

#### A. Definitions

We begin by defining an iterated source and target construction for diagrams.

**Definition 24.** For an  $n$ -diagram  $D$ , then for  $p \in \mathbb{Z}$  with  $p \leq n$ , we define the  $p$ -fold source  $D.s^p$  and  $p$ -fold target  $D.t^p$  as follows:

- if  $p > 0$  then  $D.s^p := (D.s^{p-1}).s$ ;
- if  $p \leq 1$  then  $D.s^p := D.s$ .

Note that in particular we allow  $p < 0$ , which makes certain statements below easier.

**Definition 25.** Given an  $n$ -diagram  $D$  and an  $m$ -diagram  $S$  with  $S.t^{m-n+1} = D.s^{n-m+1}$ , then if  $n > m$  we define the inclusion embedding  $\text{Inc}_r(S, D) : D \hookrightarrow S \circ D$  as follows:

$$\bullet \text{Inc}_r(S, D).h = D.\text{id}.h \quad (9)$$

$$\bullet \text{Inc}_r(S, D).e = \text{Inc}_r(S, D.s) \quad (10)$$

If  $n < m$  we define the  $\text{Inc}_l(S, D) : S \hookrightarrow S \circ D$  as follows:

$$\bullet \text{Inc}_l(S, D).h = S.\text{id}.h \quad (11)$$

$$\bullet \text{Inc}_l(S, D).e = \text{Inc}_r(S.s, D) \quad (12)$$

We present a recursive definition of composition of an  $n$ -diagram  $D$  and an  $m$ -diagram  $S$ . If  $n > m$  we specify all the generators and embeddings of the composite and then we refer recursively to the  $(n-1)$ -dimensional source of  $D$  and to the  $m$ -diagram  $S$ . That way with each recursive call  $n$  decreases, hence  $n - m$  decreases. Eventually, we decrease  $n$  sufficiently that  $n = m$ , and the recursion terminates with the base clause. The case for  $m > n$  is analogous. This ensures that the definition is well-founded.

**Definition 26.** Given an  $n$ -diagram  $D$  and an  $m$ -diagram  $S$  with  $S.t^{m-n+1} = D.s^{n-m+1}$ , the composite  $\text{max}(n, m)$ -diagram  $S \circ D$  is defined as follows. If  $n = m$ :

$$\bullet (S \circ D).s = S.s \quad (13)$$

$$\bullet |S \circ D| = |D| + |S| \quad (14)$$

$$\bullet (S \circ D)[j].g = \begin{cases} S[j].g & 0 \leq j < |S| \\ D[j-|S|].g & |S| \leq j < |D|+|S| \end{cases} \quad (15)$$

$$\bullet (S \circ D)[j].e = \begin{cases} S[j].e & 0 \leq j < |S| \\ D[j-|S|].e & |S| \leq j < |D|+|S| \end{cases} \quad (16)$$

If  $n > m$ :

$$\bullet (S \circ D).s = S \circ D.s \quad (17)$$

$$\bullet |S \circ D| = |D| \quad (18)$$

$$\bullet (S \circ D)[j].g = D[j].g \quad (19)$$

$$\bullet (S \circ D)[j].e = \text{Inc}_r(S, D[j].d) \circ D[j].d \quad (20)$$

If  $n < m$ :

$$\bullet (S \circ D).s = S.s \circ D \quad (21)$$

$$\bullet |S \circ D| = |S| \quad (22)$$

$$\bullet (S \circ D)[j].g = S[j].g \quad (23)$$

$$\bullet (S \circ D)[j].e = \text{Inc}_l(S[j].d, D) \circ S[j].e \quad (24)$$

#### B. Results

We show that the constructions we make satisfy a large number of attractive properties.

**Proposition 27** (Identity rewrites). *Given an  $n$ -diagram  $D$  and  $e : S \hookrightarrow D$ , then  $D.\Pi[e, S] = D$ .*

**Proposition 28** (Composite lifts). *For  $n \geq 0$ , given well-defined  $n$ -diagrams  $S, T, A, B, C$  with  $S.s = T.s$ ,  $S.t = T.t$ ,  $A.s = C.s$ ,  $A.t = C.t$ , and given well-defined embeddings  $e : S \hookrightarrow A$ ,  $f : C \hookrightarrow B$ , the following holds:*

$$(f.\Lambda[A] \circ e).\Lambda[T] = (f.\Lambda[A.\Pi[e, T]]) \circ e.\Lambda[T]$$

**Proposition 29** (Composite rewrites). *For  $n \geq 0$ , given well-defined  $n$ -diagrams  $S, T, A, B, C$  with  $S.s = T.s$ ,  $S.t = T.t$ ,  $A.s = C.s$ ,  $A.t = C.t$ , and given well-defined embeddings  $e : S \hookrightarrow A$ ,  $f : C \hookrightarrow B$ , the following holds for  $0 \leq j \leq e.h$ :*

$$B.\Pi[f, A.\Pi[e, T]] = B.\Pi[f, A].\Pi[f.\Lambda[A] \circ e, T]$$

**Proposition 30** (Associative composite embeddings). *For  $n \geq 0$ , given three well-defined  $n$ -diagram embeddings  $e : S \hookrightarrow D$ ,  $f : D \hookrightarrow M$ ,  $g : M \hookrightarrow N$ , the following equality holds:*

$$g \circ (f \circ e) = (g \circ f) \circ e$$

**Proposition 31** (Well-behaved whiskering). *For  $n, m \geq 0$  with  $n \neq m$ , given a well-defined  $n$ -diagram  $D$  and well-defined  $m$ -diagram  $S$  such that the composite  $S \circ D$  exists, then:*

- if  $n > m$  then  $(S \circ D)[i].d = S \circ (D[i].d)$  for any  $0 \leq i < |D|$ ;
- if  $n < m$  then  $(S \circ D)[i].d = (S[i].d) \circ D$  for any  $0 \leq i < |S|$ .

**Proposition 32** (Interaction of lifts and inclusions). *For  $n, m \geq 0$  with  $n \neq m$ , given a well-defined  $n$ -diagram  $D$  and a well-defined diagram  $m$ -diagram  $S$ , then:*

- if  $n > m$  then  $\text{Inc}_r(S, D[i].d) = (\text{Inc}_r(S, D).e).\Lambda[D[i].d]$  for any  $0 \leq i < |D|$ ;
- if  $n < m$  then  $\text{Inc}_l(S[i].d, D) = (\text{Inc}_l(S, D).e).\Lambda[S[i].d]$  for any  $0 \leq i < |S|$ .

Furthermore, the notion of diagram composition that results has properties that familiar properties expected of globular higher categories.

**Theorem 33.** *Given three well-defined diagrams: an  $n$ -diagram  $D$ ,  $m$ -diagram  $S$  and an  $l$ -diagram  $M$ , let  $a = \min(n, l) - 1$ ,  $b = \min(m, \max(n, l)) - 1$ ,  $c = \min(m, n) - 1$  and  $d = \min(\max(m, n), l) - 1$ , then provided that these composites exist, the following hold:*

$$S \circ (D \circ M) = (S \circ D) \circ M \quad \text{if } a = b \quad (25)$$

$$S \circ (D \circ_a M) = (S \circ D) \circ (SM) \quad \text{if } b < a \quad (26)$$

$$(S \circ D) \circ_d M = (S \circ M) \circ (D \circ_d M) \quad \text{if } d < c \quad (27)$$



#### IV. GRAPHICAL FORMALISM

Here we sketch an informal graphical calculus for diagram structures, following the ideas of Trimble [22]. In particular, we introduce the new idea of  $k$ -projected diagram, in which only the top  $k$  dimensions of a diagram are depicted. However, we do not present solutions to the technical difficulties that Trimble encounters, and there is substantial work still required to formalize these ideas and prove correctness and completeness of the approach.

**General procedure.** We provide a method of translating a diagram structure into a graphical representation. While we state this in arbitrary dimension, it is only precise up to dimension 3.

In a  $k$ -projected graphical representation of an  $n$ -diagram  $D$ , each  $p$ -cell in  $D$  for  $k \leq p \leq n$  is represented by an  $(n-p)$ -dimensional subspace of  $\mathbb{R}^k$ . The  $p$ -cells for  $p < k$  are not depicted. For  $k < n$ , the  $k$ -projected representation of an  $n$ -diagram does not provide complete information about that diagram. Nonetheless, it is the correct notion for describing the action of homotopy generators, as we explore in Section V.

**Definition 34.** For an  $n$ -diagram  $D$  over a signature  $\Sigma$ , for  $k \leq n$ , its  $k$ -projected graphical representation  $G_D^k \subset \mathbb{R}^k$  is a labelled partitioned subspace, defined as follows:

- for  $n = 0$ , we have  $G_D^0 := \{\bullet\} \subseteq \mathbb{R}^0$ ;
- for  $n > 0$ :
  - ★ at height  $i$ , to agree with  $G_{D[i].d}^{k-1} \subset \mathbb{R}^{n-1} \subset \mathbb{R}^n$ ;
  - ★ between heights  $i$  and  $i+1$ , as a glued double cone, with centre point labelled by the element  $D[i].g \in \Sigma_n$ .

The gluing scheme here is straightforward in low dimensions, but not precise in general. We illustrate it here by example.

**Example.** Consider the following 2-signature  $\Sigma$ , with some components coloured to help identification:

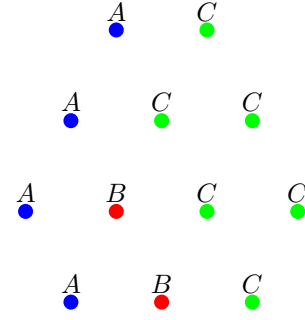
<b>0-cells</b>	$A, B, C$
<b>1-cells</b>	$F : A \rightarrow B$ $G : B \rightarrow C$ $H : C \rightarrow C$
<b>2-cells</b>	$\mu : H \Rightarrow G \circ F$ $\nu : F \circ G \Rightarrow F$ $\phi : H \Rightarrow \text{id}_C$

Then consider the following 2-diagram  $D$  defined over  $\Sigma$ :

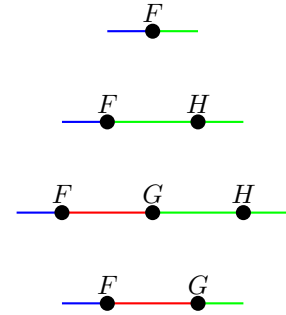
$D.s[0].g = A$	
$D.s[0].g = F$	$D.s[0].e = []$
$D.s[1].g = F$	$D.s[1].e = []$
$D[0].g = \mu$	$D[0].e = [1]$
$D[1].g = \nu$	$D[1].e = [0]$
$D[2].g = \phi$	$D[2].e = [1]$

Note that according to Definition 9, embeddings between 0-diagrams consist of no data. For this reason the lists  $D.s[0].e$  and  $D.s[1].e$  are empty.

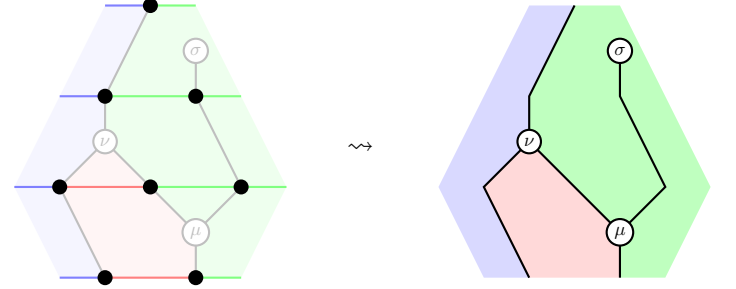
Following the recursive procedure, we begin at the base cases, depicting  $A$ ,  $B$  and  $C$  as points as follows:



One step up the call stack, we represent the 1-diagram slices  $D[0].d$ ,  $D[1].d$ ,  $D[2].d$  and  $D[3].d$  as 1-dimensional glued cones over these points, labelling the centre point of the cones with the name of the associated generator:



Then at the top level, we glue 2-dimensional cones between the 1-dimensional slices to produce the 2-dimensional image:



This idea can be continued straightforwardly into dimension 3, yielding representations of 3-projected  $n$ -diagrams for  $n \geq 3$  as subspaces of  $\mathbb{R}^3$ .

**Movies.** Another way to visualize an  $n$ -diagram  $D$  is by visualizing each  $(n-1)$ -dimensional slice  $D[i].d$  for  $0 \leq i \leq |D|$  sequentially. We call this a *movie*. By iterating this idea, one can visualize diagrams of arbitrary dimension without using projections. We demonstrate this in Section A with a movie presentation of a 5-diagram, giving each slice as a 2-projection.

#### V. HOMOTOPY GENERATORS

In this section we introduce the homotopy generators which must be supported by signatures as part of the definitions of semistrict  $n$ -category, for  $2 \leq n \leq 4$ , which we give in

the introduction. We also describe their extended versions, required for the definition of quasistrict 4-category.

**Invertibility.** We use a familiar coinductive definition of invertible cell in a higher category [54]. Given a generator  $f$ , we write  $[f]$  for the diagram of height 1 consisting only of that generator; and for an  $n$ -diagram  $D$ , we write  $D.\text{Id}$  for the identity  $(n+1)$ -diagram on  $D$ .

**Definition 35.** For  $n > 0$ , given an  $n$ -signature  $\Sigma$ , a  $k$ -generator  $f \in \Sigma_k$  for  $0 < k \leq n$  is *invertible* when it is equipped with:

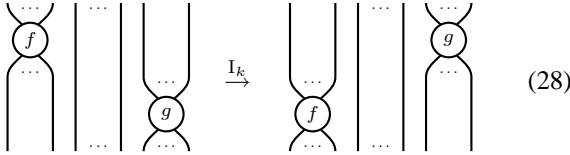
- an *inverse*  $f^{-1} \in \Sigma_k$  with  $f^{-1}.t = f.s$  and  $f^{-1}.s = f.t$ ;
- when  $k < n$ , invertible  $(k+1)$ -cells  $f', f'' \in \Sigma_{k+1}$  as follows:

$$\begin{aligned} f'.s &= [f] \circ [f^{-1}] & f''.s &= f.t.\text{Id} \\ f'.t &= f.s.\text{Id} & f''.t &= [f^{-1}] \circ [f] \end{aligned}$$

#### A. Type I homotopy generators

This move allows height rearrangement of adjacent cells. Note that these generators are always of dimension 3 or higher.

**Definition 36** (Type I homotopy generator). For  $n \geq 0$ , an  $n$ -signature  $\Sigma$  supports *type I homotopy generators* if for any  $2 \leq k < n$ , for any  $k$ -diagram with 2-projection given by the left-hand diagram below, there is a chosen invertible  $(k+1)$ -cell  $I_k$  as follows:


(28)

By abuse of notation, we simply refer to these  $(k+1)$ -cells as  $I_k$ . They are indexed formally by their entire source diagram, drawn here on the left; while the left-hand diagram may be a  $k$ -diagram for  $k \geq 2$ , only the features visible in its 2-projection are relevant for constructing the associated type I homotopy generator. Since they are invertible, the reverse rewrite is also allowed. We emphasize some key features of the left-hand diagram:

- there are no wires to the left of  $f$  or right of  $g$ ;
- there may be wires to the right of  $f$  and left of  $g$ ;
- $f$  and  $g$  have arbitrary input and output wires;
- $f$  and  $g$  are generators, not composite diagrams.

Sometimes we depict type I homotopy generators as single 3-projected diagrams using a braiding convention, as illustrated in Figure 4. This braiding convention is an artistic style, and we do not attempt to formalize it. Note also that for clarity we omit the intervening sheets between  $f$  and  $g$ .

**Expansion scheme.** We require an expansion scheme for type I homotopy generators, defining their action on composite diagrams. We provide this scheme recursively.

**Definition 37** (Type I homotopy composite). In an  $n$ -signature that supports type I homotopy generators, a *type I homotopy composite* is a diagram formed from a sequence of type

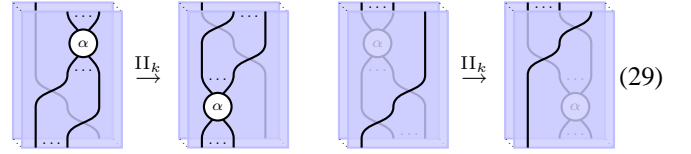
I homotopy generators, with values defined recursively as illustrated in Figure 6.

Note that any signature supporting type I homotopy generators always supports type I homotopy composites.

#### B. Type II homotopy generators

Homotopy generators of type II are defined in the following way. Note that these generators are always of dimension 4 or higher.

**Definition 38** (Type II homotopy generator). For  $n \geq 0$ , given an  $n$ -signature  $\Sigma$  supporting type I homotopy generators,  $\Sigma$  supports *type II homotopy generators* if for any  $3 \leq k < n$ , for  $k$ -diagrams with 3-projections given by the source diagrams below for  $\alpha, \beta \in \Sigma_k$ , there are chosen invertible  $(k+1)$ -cells  $\Pi_k$  as follows:


(29)

Note that these diagrams involve homotopy generators of type I and their expansion scheme. As per the convention set out above, these diagrams may have arbitrary interposing sheets, but we omit them for notational clarity. However, it is deliberate that there are no additional sheets in front or behind the displayed diagrams.

**Expansion scheme.** We require expansion schemes for type II homotopy generators, defining their action on composite diagrams.

**Definition 39.** In an  $n$ -signature that supports type II homotopy generators, a *type II homotopy composite* is a diagram formed from a sequence of type II homotopy generators, denoted  $\tilde{\Pi}$ , with values defined recursively in Figure 7.

A similar composition scheme is needed for the other style of type II homotopy generator, where the vertex is on the rear sheet. As with type I homotopy generators, any signature supporting type II homotopy generators always supports type II homotopy composites.

**Extended variant.** The type  $\Pi'$  homotopy generators include 2 further variants of the moves shown in (29), corresponding to

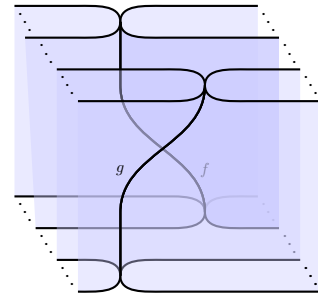


Fig. 4: A 3-projected diagram of a type I homotopy move.

pulling a vertex through an inverse type I homotopy generator on the front or rear sheet, as well as the corresponding expansion schemes.

### C. Type III homotopy generators

We define type III homotopy generators as follows. Note that these generators are always of dimension 5 or higher.

**Definition 40** (Type III homotopy generator). For  $n \geq 0$ , given an  $n$ -signature  $\Sigma$  supporting type II homotopy generators,  $\Sigma$  supports *type III homotopy generators* if for any  $4 \leq k < n$ , for  $k$ -diagrams with 4-projections given by the source diagrams below, there are chosen invertible  $(k+1)$ -cells  $\text{III}_k$  as shown in Figure 8.

These diagrams make use of the expansion scheme for type II homotopy generators, since  $\mu.s$  and  $\mu.t$  are diagrams, not generators.

**Expansion scheme.** We have no need for an expansion scheme for type III homotopy generators. Such a scheme would be needed for the definition of semistrict 5-category.

**Extended variant.** The type  $\text{III}'$  homotopy generators include 2 further variants of the moves shown in Figure 8, corresponding to the extended type  $\text{II}'$  homotopy generators.

### D. Type IV homotopy generators

We define type IV homotopy generators as follows. Note that these generators are always of dimension 5 or higher.

**Definition 41** (Type IV homotopy generator). For  $n \geq 0$ , given an  $n$ -signature  $\Sigma$  supporting type II homotopy generators,  $\Sigma$  supports *type IV homotopy generators* if for any  $4 \leq k < n$ , for a  $k$ -diagram with 4-projections given by the counterclockwise diagram below, there are chosen invertible  $(k+1)$ -cells  $\text{IV}_k$  as follows:

$$\begin{array}{ccc}
 \text{Diagram 1} & \begin{array}{c} \xrightarrow{\text{II}_{k-1}} \\ \uparrow \text{IV}_k \\ \xrightarrow{\text{II}_{k-1}^{-1}} \end{array} & \text{Diagram 2}
 \end{array} \quad (30)$$

In the clockwise path, the upper crossing is ‘pulled down’ by the  $\text{II}_{k-1}$  move. In the counterclockwise path, the lower crossing is ‘pulled up’ by the  $\text{II}_{k-1}^{-1}$  move. Both of these can be seen as ways to interpret the third Reidemeister move. The type IV homotopy generator says that these two diagrams are related by an invertible cell.

The need for this homotopy was identified by Breen [55] and the cell has been referred to in the literature as the ‘Breenator’.

**Expansion scheme.** We have no need for an expansion scheme for type IV homotopy generators.

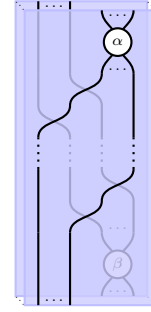
**Extended variant.** The type  $\text{IV}'$  homotopy generators include 2 further variants of the moves shown in (30), corresponding to the inverse type I homotopy generators.

### E. Type V homotopy generators

We now describe type V homotopy generators. These generators are always of dimension 5 or higher. They require a rearrangement scheme for type I homotopy generators, labelled  $\hat{\text{I}}$ ; this is described below.

**Definition 42** (Type V homotopy generator). For  $n \geq 0$ , given an  $n$ -signature  $\Sigma$  supporting type II homotopy generators,  $\Sigma$  supports *type V homotopy generators* if for any  $4 \leq k < n$ , a  $k$ -diagram with 4-projections given by the counterclockwise diagram below for  $\alpha, \beta \in \Sigma_{k-1}$ , there are chosen invertible  $(k+1)$ -cells  $\text{V}_k$  as shown in Figure 9.

**Type I homotopy generator rearrangement scheme.** The type V homotopy generators require a rearrangement scheme for the type I homotopy generators. To see why this is required, consider the picture below:



The generator  $\beta$  can be ‘pulled up’ by a  $\text{II}^{-1}$  move, but the generator  $\alpha$  cannot be ‘pulled down’ by a  $\text{II}$  move, since the type I generators, drawn as crossings, are not correctly arranged. To remedy this, we need a rearrangement scheme for type I homotopy generators. We denote this scheme  $\hat{\text{I}}$ , and construct it recursively as follows.

**Definition 43.** In an  $(n+1)$ -signature  $\sigma$  that supports interchangers of type  $\text{I}_k$ , a *rearrangement*  $\hat{\text{I}}_k$  consists of a sequence of applications of the composite interchanger of type  $\hat{\text{I}}_k$ , with values defined recursively as shown in Figure 10.

**Expansion scheme.** We do not need an expansion scheme for type V homotopy generators.

**Extended variant.** The type  $\text{V}'$  homotopy generators include 2 further variants of the moves shown in (29), corresponding to pulling a vertex through an inverse type I homotopy generator, as well as the corresponding composition schemes.

### F. Type VI homotopy generators

These are required for the definition of quasistrict 4-category. They involve reflected versions of the move illustrated in Figure 11.

## REFERENCES

- [1] T. Univalent Foundations Program, *Homotopy Type Theory: Univalent Foundations of Mathematics*. Institute for Advanced Study: <https://homotopytypetheory.org/book>, 2013.
- [2] S. Awodey and M. A. Warren, “Homotopy theoretic models of identity types,” *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 146, no. 01, p. 45, 2008.

- [3] V. Voevodsky, “A very short note on the homotopy  $\lambda$ -calculus,” 2006, [Available online](#).
- [4] M. Hofmann and T. Streicher, “The groupoid model refutes uniqueness of identity proofs,” in *LICS 1994*. IEEE, 1994.
- [5] J. Lurie, “On the classification of topological field theories,” *Current Developments in Mathematics*, vol. 2008, no. 1, pp. 129–280, 2008.
- [6] J. C. Baez and J. Dolan, “Higher-dimensional algebra and topological quantum field theory,” *JMP*, vol. 36, no. 11, p. 6073, 1995.
- [7] M. Atiyah, “Topological quantum field theories,” in *The geometry and physics of knots*. CUP, 2009, pp. 12–23.
- [8] C. Schommer-Pries, “The Classification of Two-dimensional Extended Topological Field Theories,” Ph.D. dissertation, Department of Mathematics, University of California, Berkeley, 2009.
- [9] J. Lurie, *Higher Topos Theory (AM-170)*. Walter de Gruyter GmbH, 2009.
- [10] A. Grothendieck, “Pursuing stacks,” 1983, [Available online](#).
- [11] R. Bruni, J. Meseguer, and U. Montanari, “Symmetric monoidal and cartesian double categories as a semantic framework for tile logic,” *MSCS*, vol. 12, no. 01, 2002.
- [12] E. Goubault, “Some geometric perspectives in concurrency theory,” *Homology, Homotopy and Applications*, vol. 5, no. 2, pp. 95–136, 2003.
- [13] S. Mimram, “Towards 3-dimensional rewriting theory,” *LMCS*, vol. 10, no. 2, 2014.
- [14] Y. Lafont, “Two-dimensional rewriting,” in *Rewriting Techniques and Applications*, 1997, pp. 228–229.
- [15] Y. Guiraud and P. Malbos, “Higher-dimensional normalisation strategies for acyclicity,” *Adv. Math.*, vol. 231, no. 3–4, pp. 2294–2351, 2012.
- [16] D. Reutter and J. Vicary, “Biunitary constructions in quantum information,” 2016.
- [17] A. Jaffe, Z. Liu, and A. Wozniakowski, “Holographic software for quantum networks,” 2016.
- [18] V. authors, “Strict 2-category,” entry on the *nLab*. [Online]. Available: <https://ncatlab.org/nlab/show/strict+2-category>
- [19] C. L. Douglas and A. G. Henriques, “Internal bicategories,” 2012.
- [20] S. Crans, “On braidings, syllepses and symmetries,” *Cahiers de Topologie et Geometrie Differentielle Categoriques*, vol. XLI, no. 1, pp. 2–74, 1998.
- [21] T. Leinster, *Higher Operads, Higher Categories*. CUP, 2004.
- [22] T. Trimble, “Surface diagrams,” 1995, [nLab:Surface diagrams](#).
- [23] A. Joyal and R. Street, “The geometry of tensor calculus I,” *Adv. Math.*, vol. 88, pp. 55–112, 1991.
- [24] J. W. Barrett, C. Meusburger, and G. Schaumann, “Gray categories with duals and their diagrams,” 2012.
- [25] R. Street, “Low-dimensional Topology and Higher-order Categories,” 1995.
- [26] P. Selinger, “A survey of graphical languages for monoidal categories,” *New Structures for Physics*, 2009.
- [27] J. Baez and M. Neuchl, “Higher dimensional algebra I: Braided monoidal 2-categories,” *Adv. Math.*, vol. 121, no. 2, pp. 196–244, 1996.
- [28] M. Batanin, “Monoidal globular categories as a natural environment for the theory of weak  $n$ -categories,” *Adv. Math.*, vol. 136, no. 1, pp. 39–103, 1998.
- [29] B. Bartlett, “Quasistrict symmetric monoidal 2-categories via wire diagrams,” 2014.
- [30] C. Simpson, “Homotopy types of strict 3-groupoids,” 1998.
- [31] K. Bar, A. Kissinger, and J. Vicary, “Globular: a proof assistant for higher-dimensional rewriting,” in *Proc. FSCD 2016*, vol. 52, no. 34, 2016, pp. 1–11.
- [32] E. Riehl and D. Verity, “Homotopy coherent adjunctions and the formal theory of monads,” *Adv. Math.*, vol. 286, pp. 802–888, 2016.
- [33] R. Gordon, J. Power, and R. Street, “Coherence for Tricategories,” *Mem. AMS*, vol. 117, no. 558, 1995.
- [34] T. Trimble, 1995. [Online]. Available: <http://math.ucr.edu/home/baez/trimble/tetracategories.html>
- [35] K. Bar and J. Vicary, “Data structures for quasistrict higher categories,” 2016.
- [36] J. W. Gray, *Formal Category Theory: Adjointness for 2-Categories*, 1974.
- [37] —, “Coherence for the tensor product of 2-categories, and braid groups,” in *Algebra, Topology, and Category Theory*. Elsevier BV, 1976, pp. 63–76.
- [38] A. E. Hoffnung, “Spans in 2-categories: A monoidal tricategory,” 2011.
- [39] “The HoTT library: A formalization of homotopy type theory in Coq, eprint = 1610.04591, author = Andrej Bauer and Jason Gross and Peter LeFanu Lumsdaine and Mike Shulman and Matthew Sozeau and Bas Spitters.”
- [40] B. Ahrens, D. Grayson, C. Lelay, and V. Voevodsky, “Unimath: Univalent mathematics,” 2016. [Online]. Available: <https://github.com/orgs/UniMath/people>
- [41] S. Mimram, “Implementing polygraphs,” presentation at CATHRE ANR meeting.
- [42] —, “Computing critical pairs in polygraphs,” 2009, paper presented at CAMCAD 2009.
- [43] S. Crans, “A tensor product for Gray-categories,” *Theory and Applications of Categories*, vol. 5, no. 2, pp. 12–69, 1999, [TAC:05.02](#).
- [44] J. Bourke and N. Gurski, “A cocategorical obstruction to tensor products of gray-categories,” *Theory and Applications of Categories*, vol. 30, no. 11, pp. 387–409, 2015, [TAC:30.11](#).
- [45] M. Weber, “Multitensors as monads on categories of enriched graphs,” *Theory and Applications of Categories*, vol. 28, no. 26, pp. 857–932, 2013, [TAC:28.26](#).
- [46] M. Batanin, D.-C. Cisinski, and M. Weber, “Multitensor lifting and strictly unital higher category theory,” *Theory and Applications of Categories*, vol. 28, no. 25, pp. 804–856, 2013, [TAC:28.25](#).
- [47] M. Weber, “Free products of higher operad algebras,” *Theory and Applications of Categories*, vol. 28, no. 2, pp. 24–65, 2013, [TAC:28.2](#).
- [48] R. Garner, “Homomorphisms of higher categories,” *Adv. Math.*, vol. 224, no. 6, pp. 2269–2311, 2010.
- [49] —, “Understanding the small object argument,” *Applied Categorical Structures*, vol. 17, no. 3, pp. 247–285, 2008.
- [50] J. C. Baez and J. Dolan, “Higher-dimensional algebra III:  $n$ -categories and the algebra of opetopes,” *Adv. Math.*, vol. 135, no. 2, pp. 145–206, 1998.
- [51] J. Kock, A. Joyal, M. Batanin, and J.-F. Mascari, “Polynomial functors and opetopes,” *Adv. Math.*, vol. 224, no. 6, pp. 2690–2737, 2010.
- [52] E. Finster, “Type theory and the opetopes,” 2012, talk presented at the Polish Seminar on Category Theory and its Applications.
- [53] —, “The *Opetopic!* proof assistant,” [opetopic.net](#).
- [54] M. Shulman, “Coinductive definitions,” 2011, post at the *n-Cat*egory Cafe.
- [55] L. Breen, “Letter to Michael Hopkins.”
- [56] D. Verity, “Enriched Categories, Internal Categories and Change of Base,” *Reprints in Theory and Applications of Categories*, no. 20, pp. 1–266, 2011.
- [57] N. Gurski, “An algebraic theory of tricategories,” Ph.D. dissertation, 2007, [Available online](#).
- [58] —, “Biequivalences in Tricategories,” *Theory and Applications of Categories*, vol. 26, no. 14, pp. 349–384, 2012.
- [59] P. Pstragowski, “On dualizable objects in monoidal bicategories,” Master’s thesis, Bonn University, 2014.



In this Appendix we give a proof of the following theorem.

**Theorem 44.** *In a quasistrict 4-category, an adjunction of 1-morphisms gives rise to a coherent adjunction satisfying the butterfly equations.*

The butterfly equations are equations holding between specified 4-cells. By general results of Riehl and Verity [32], this is expected to hold in any correct algebraic model of 4-categories. We believe that our proof is the first to be given; indeed, we believe that it is the first substantial proof of any sort to be presented in an algebraic 4-category. The full proof itself is formalized in the *Globular* proof assistant at [globular.science/1605.002](https://globular.science/1605.002), and given in substantial detail later in this section.

There is a corresponding result at the level of 3-categories, which reads as follows.

**Theorem 45.** *In a tricategory, an adjunction of 1-morphisms gives rise to a coherent adjunction satisfying the swallowtail equations.*

The swallowtail equations are equations holding between specified 3-cells. This was first established by Verity [56], and later discussed in depth by Gurski [57, 58] and Pstragowski [59]. Given an adjunction in a quasistrict 4-category, one can apply the proof of Theorem 45 (suitably expressed in a Gray category) to obtain invertible 4-morphisms called *swallowtailators*, which witness the satisfaction up to isomorphism of the swallowtail equations. The butterfly equations are certain equations that these swallowtailators may, or may not, satisfy.

Analogously to the proof of Theorem 45, the proof of Theorem 44 proceeds by redefining one of the swallowtailators in a certain way, and then demonstrating explicitly that a certain equation holds. There are two butterfly equations; the proofs of each are in fact rather different, but for reasons of length we only present one of them here.

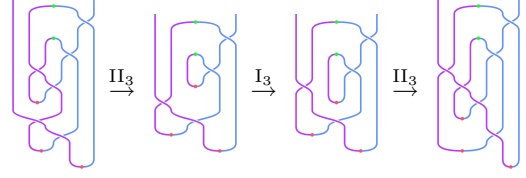
This appendix is structured as follows. In Section A-A we examine some interesting steps of the proof as rewrites of movies of 2-projected 3-diagrams, and see how they arise as instances of our homotopy generators. In Section A-B we illustrate the entire proof as a single 2-projected 5-diagram of height 140. In Section A-C we illustrate the entire proof in more detail, as a movie of 2-projected 4-diagrams; each frame in this section summarizes a movie of movies of 2-diagrams.

#### A. Proof highlights

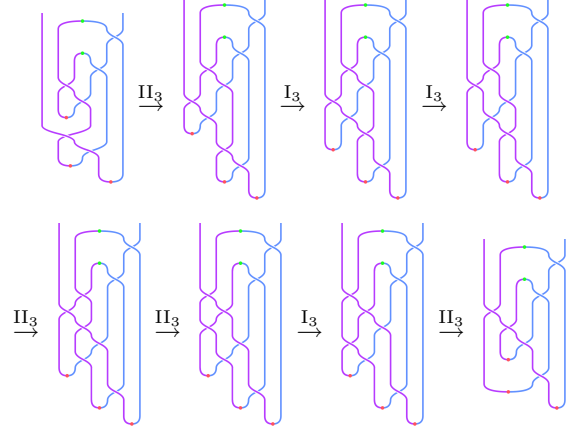
Here we visualize in detail steps 17, 83 and 97 of the main proof 5-diagram. Every step of the proof is a rewrite, and for these steps we give the source and target 4-diagrams as movies of 2-projected 3-diagrams. In each case, it can be seen how it arises as a special case of various homotopy generators.

We emphasize that every step of the main proof was produced with a single mouse interaction in the *Globular* proof assistant; the complexity we see here arises from unpacking the definitions of the homotopy moves.

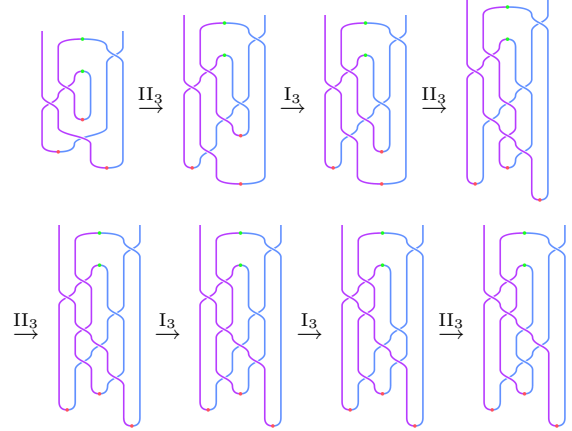
**Step 17.** This step is an instance of a type III' homotopy generator. The source has the following structure:



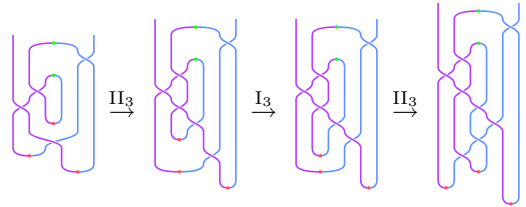
The target has the following structure:



**Step 83.** This is another instance of a type III' homotopy generator. The highlighted box of the initial slice has the following representation as a movie of 2-projected 3-diagrams:



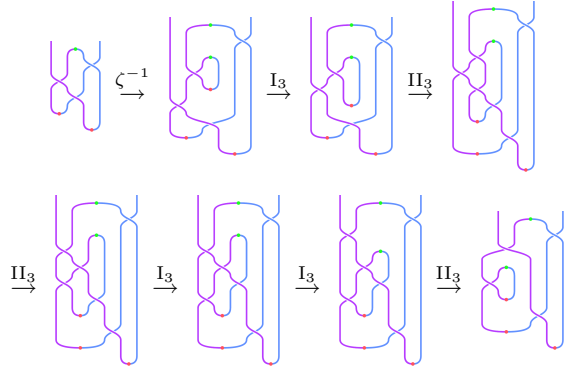
This is rewritten as follows:



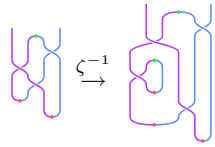
Note how the 4-cell of type  $\Pi_3$ , where the red node is pulled down and to right over the blue wire, is executed at the beginning of the first sequence, but at the end of the second sequence.



**Step 97.** Once again we draw the highlighted region of the previous slice as a movie of 2-projected 3-diagrams:

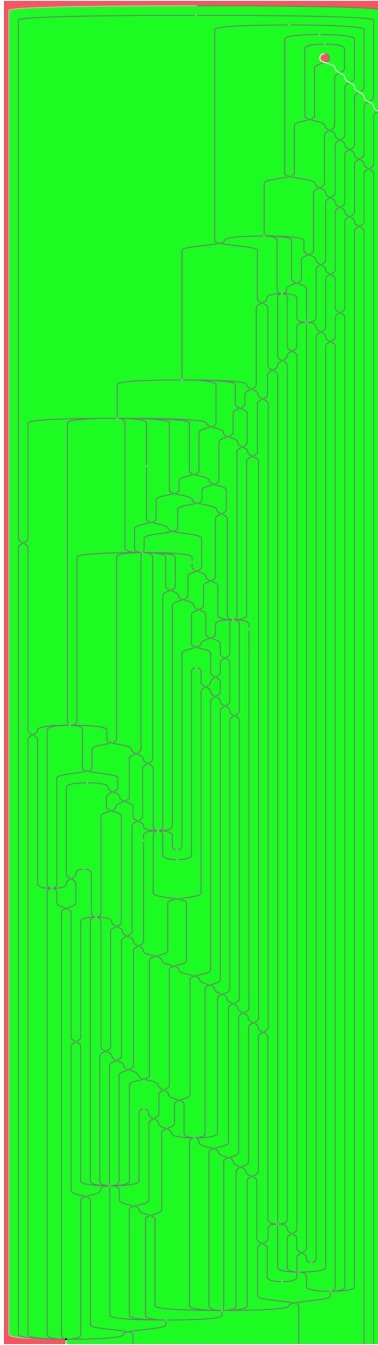


This is rewritten into the following:



*B. Projected view*

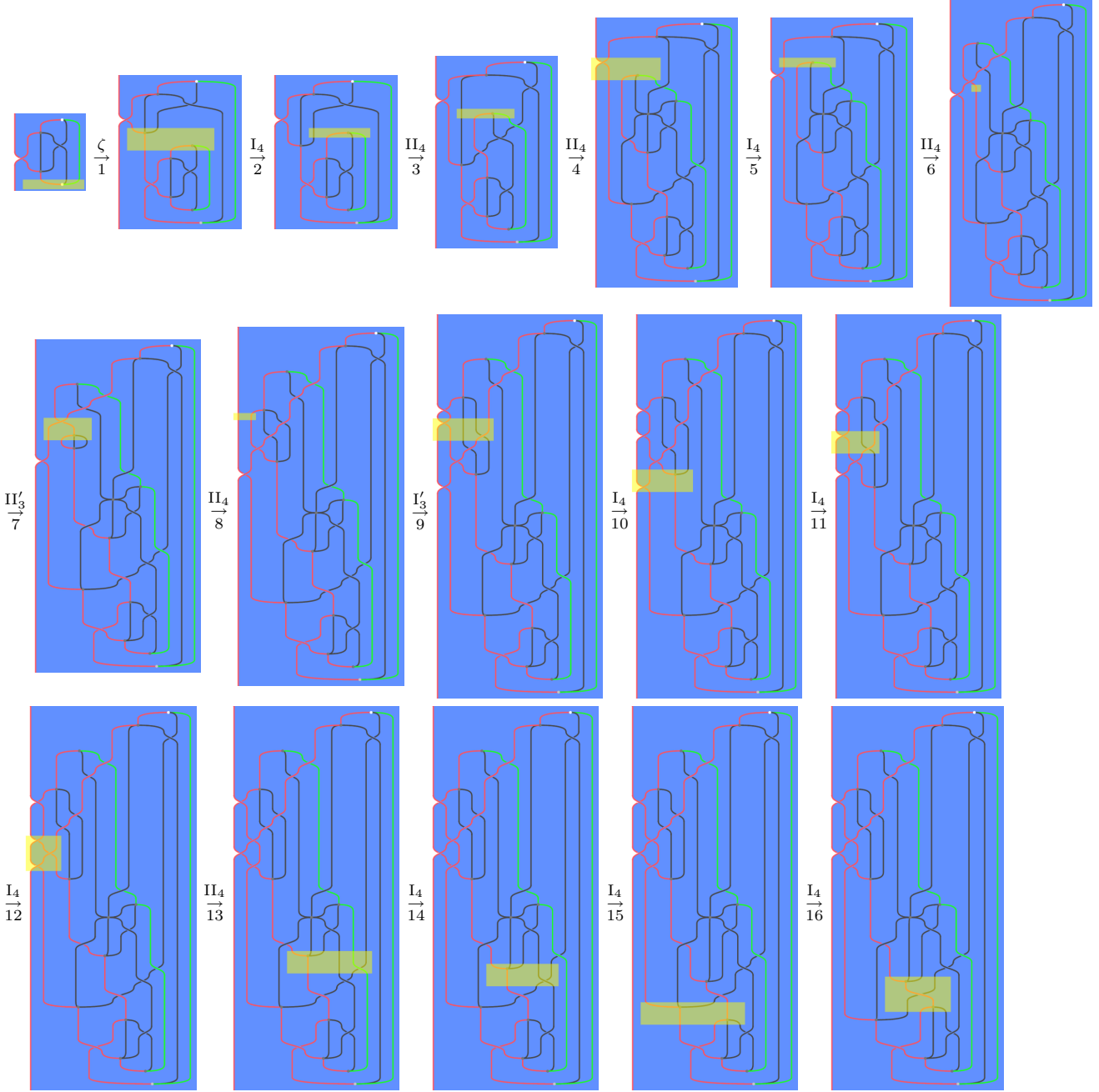
Figure 5 gives a 2-projection of the proof 5-diagram.

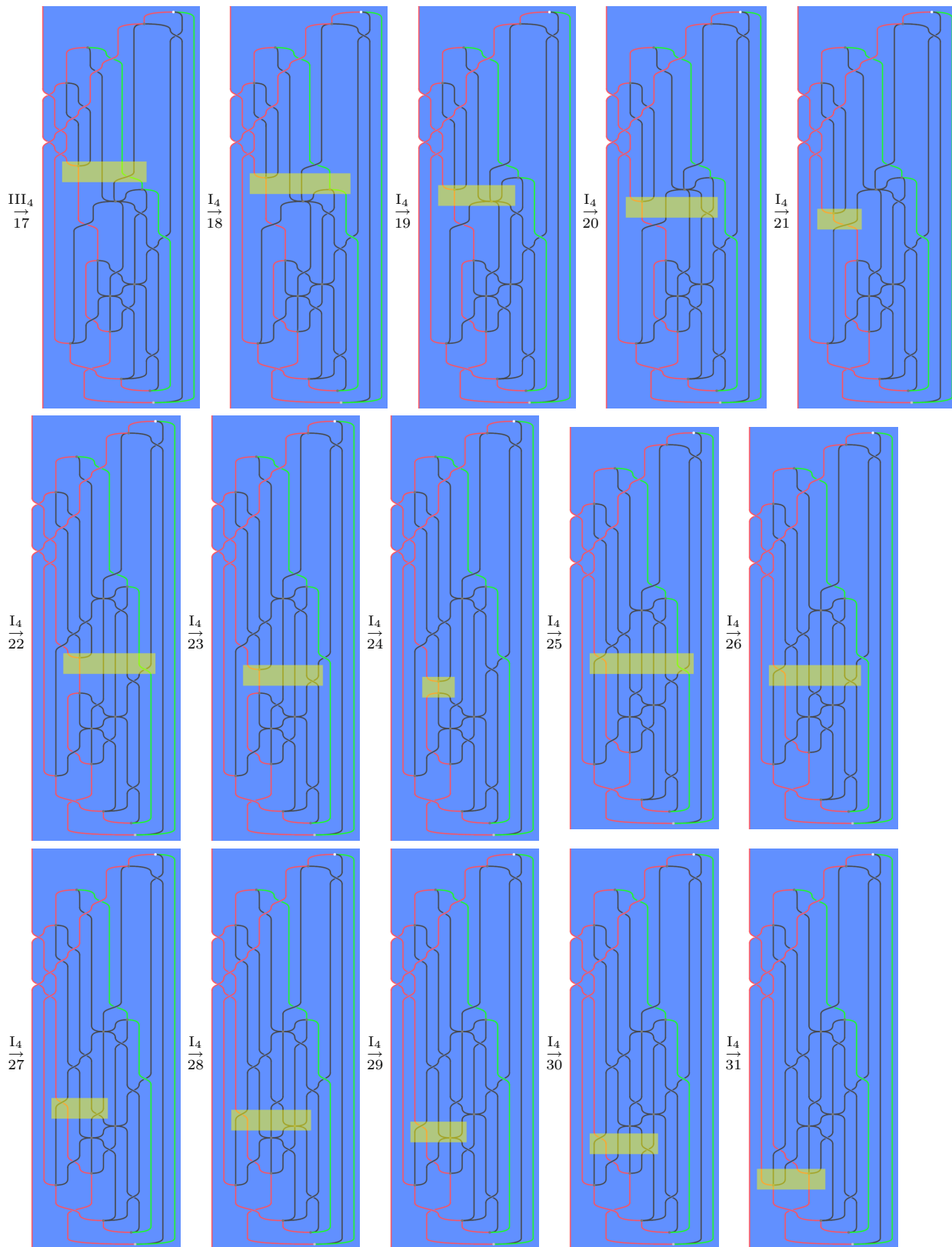


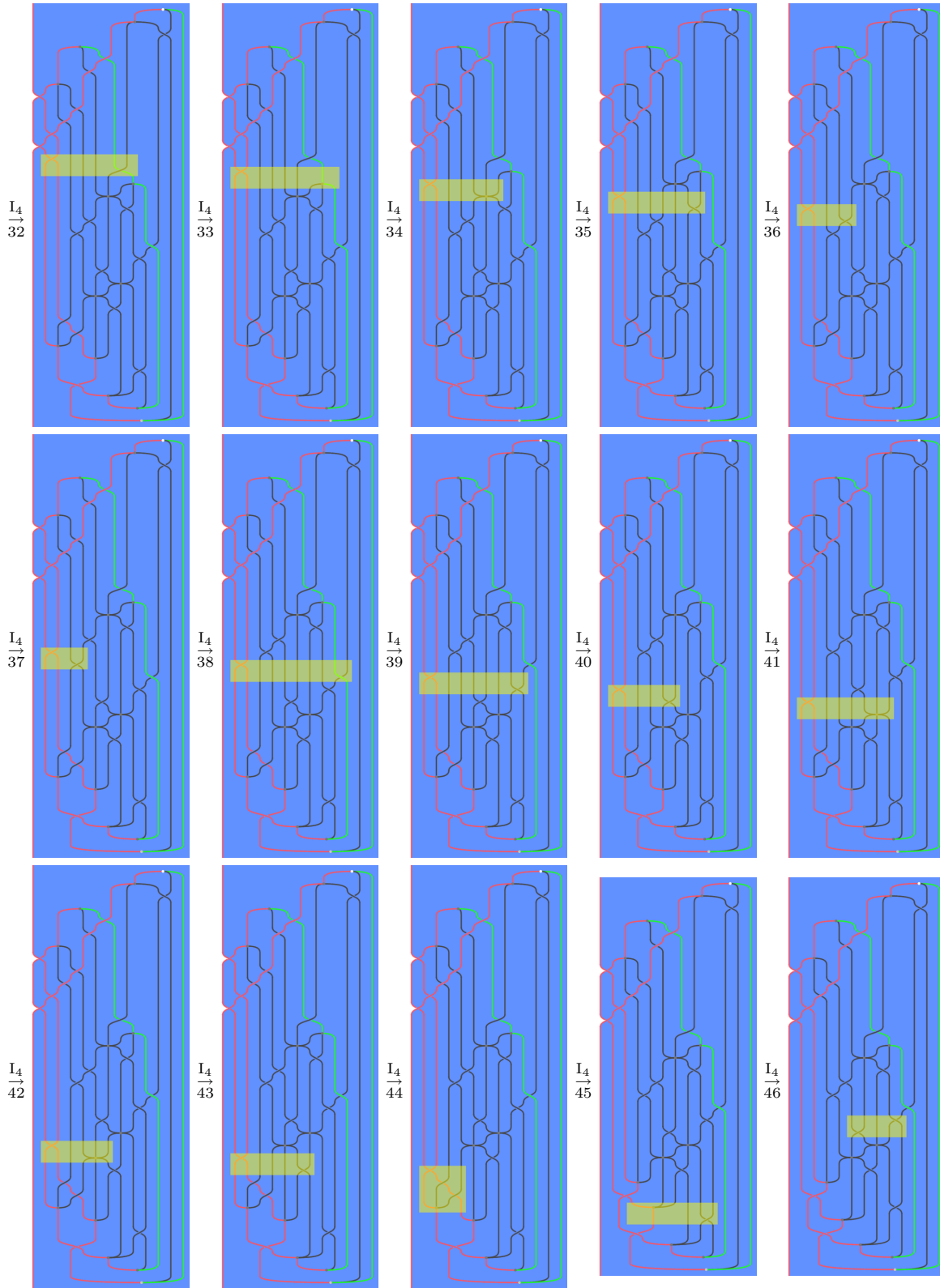
*Fig. 5: The 2-projection of the proof 5-diagram.*

### C. Movie view

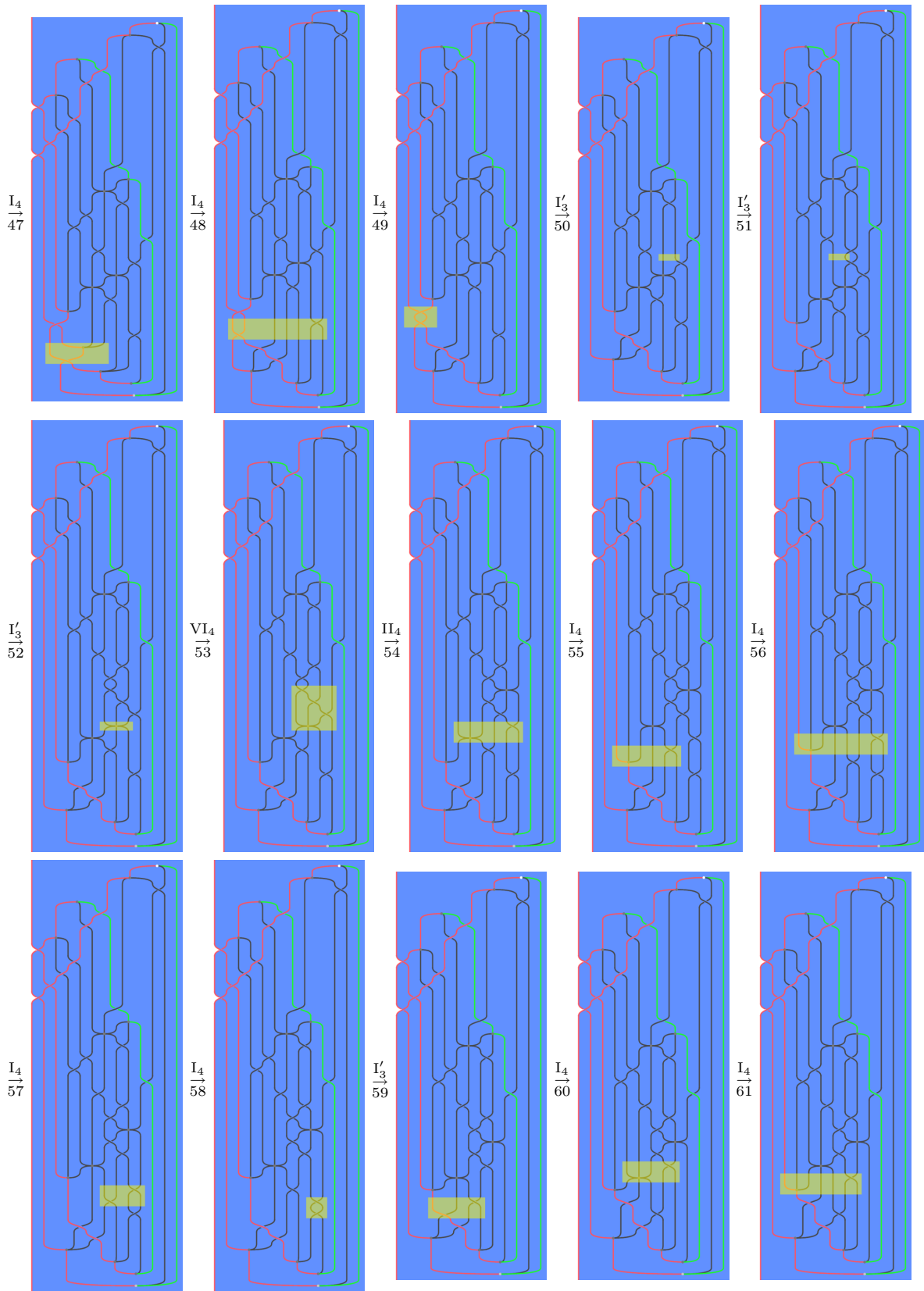
The following is a movie of 2-projections of 4-diagram slices of the proof 5-diagram. Each frame is only a partial description of that proof slice, and to be fully specified would require a movie of movies of 2-dimensional images.

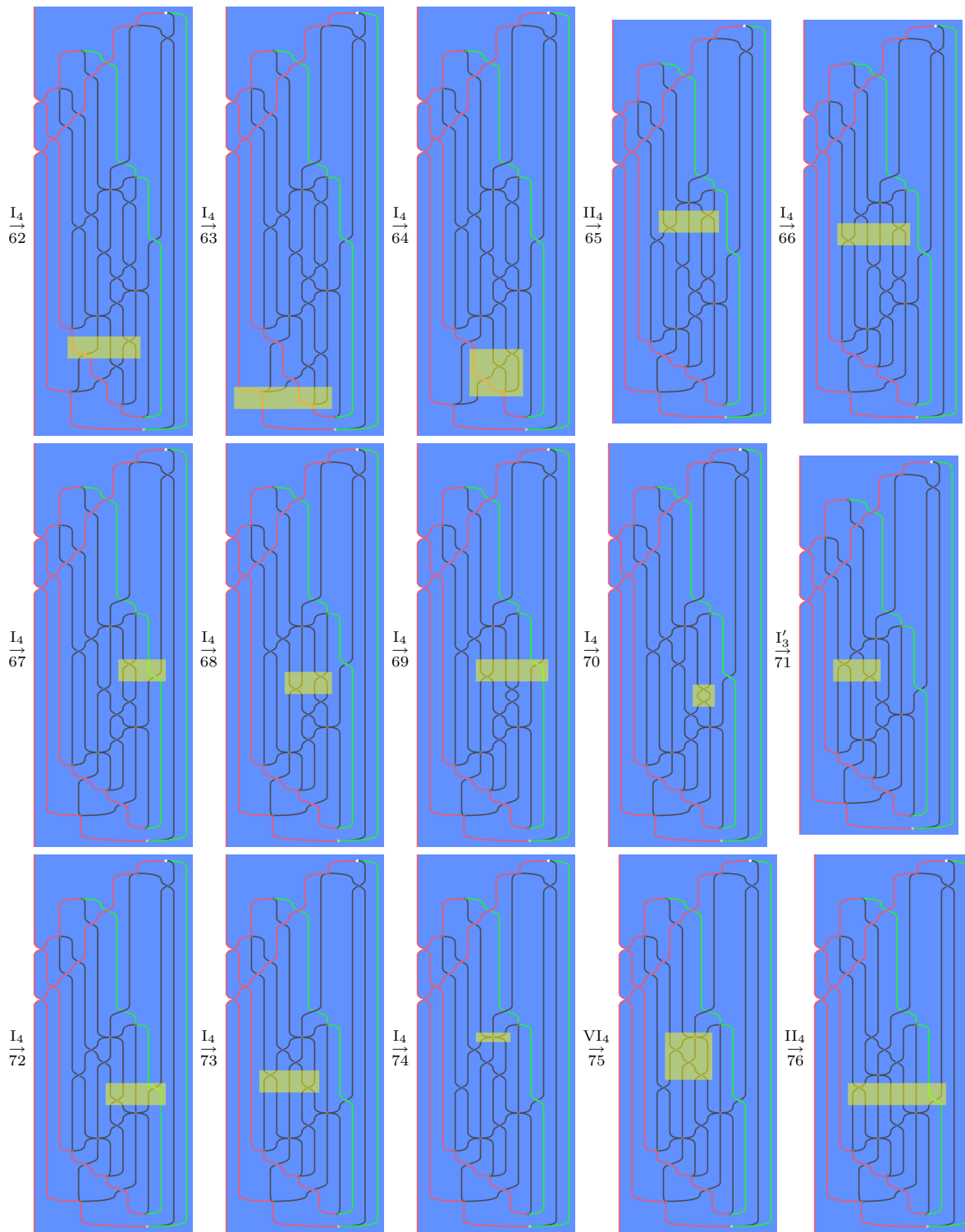


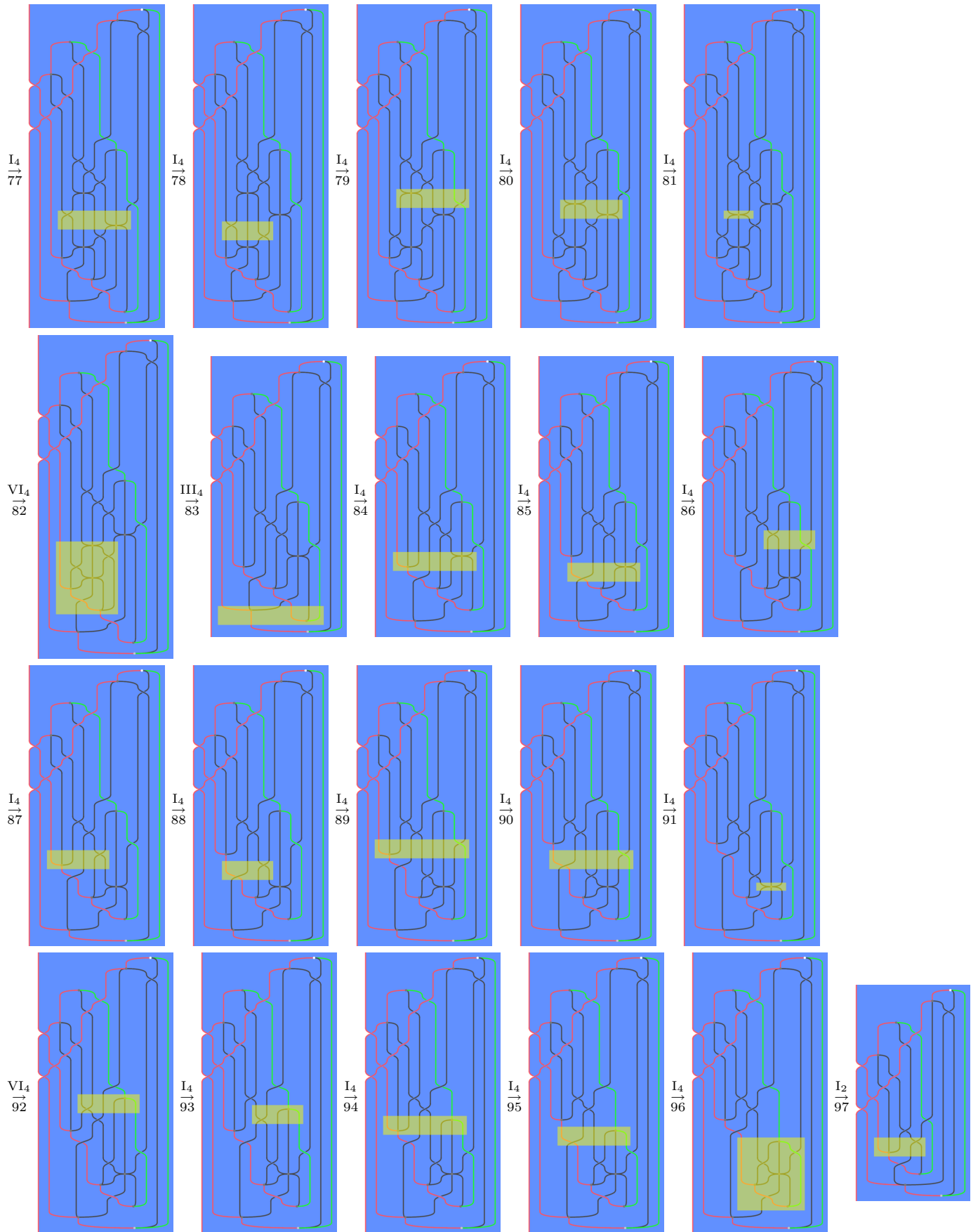


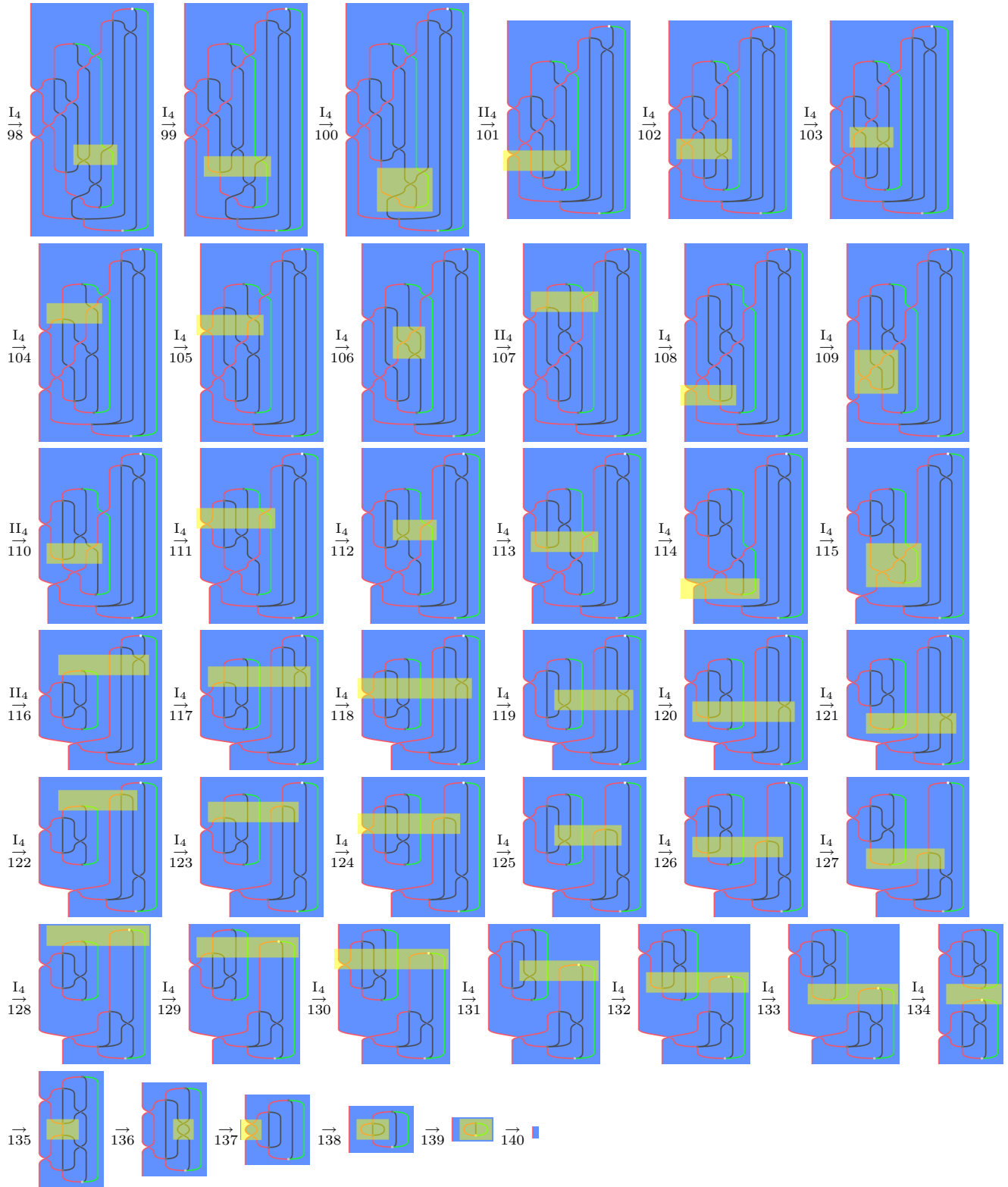












The 6 final steps are applications of the coinductive definition of invertibility.

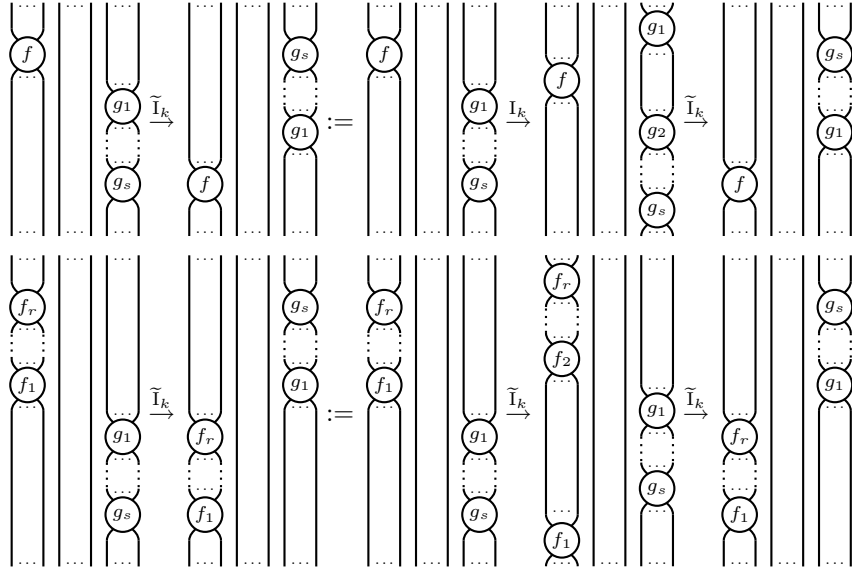


Fig. 6: Expansion scheme for type I homotopy moves.



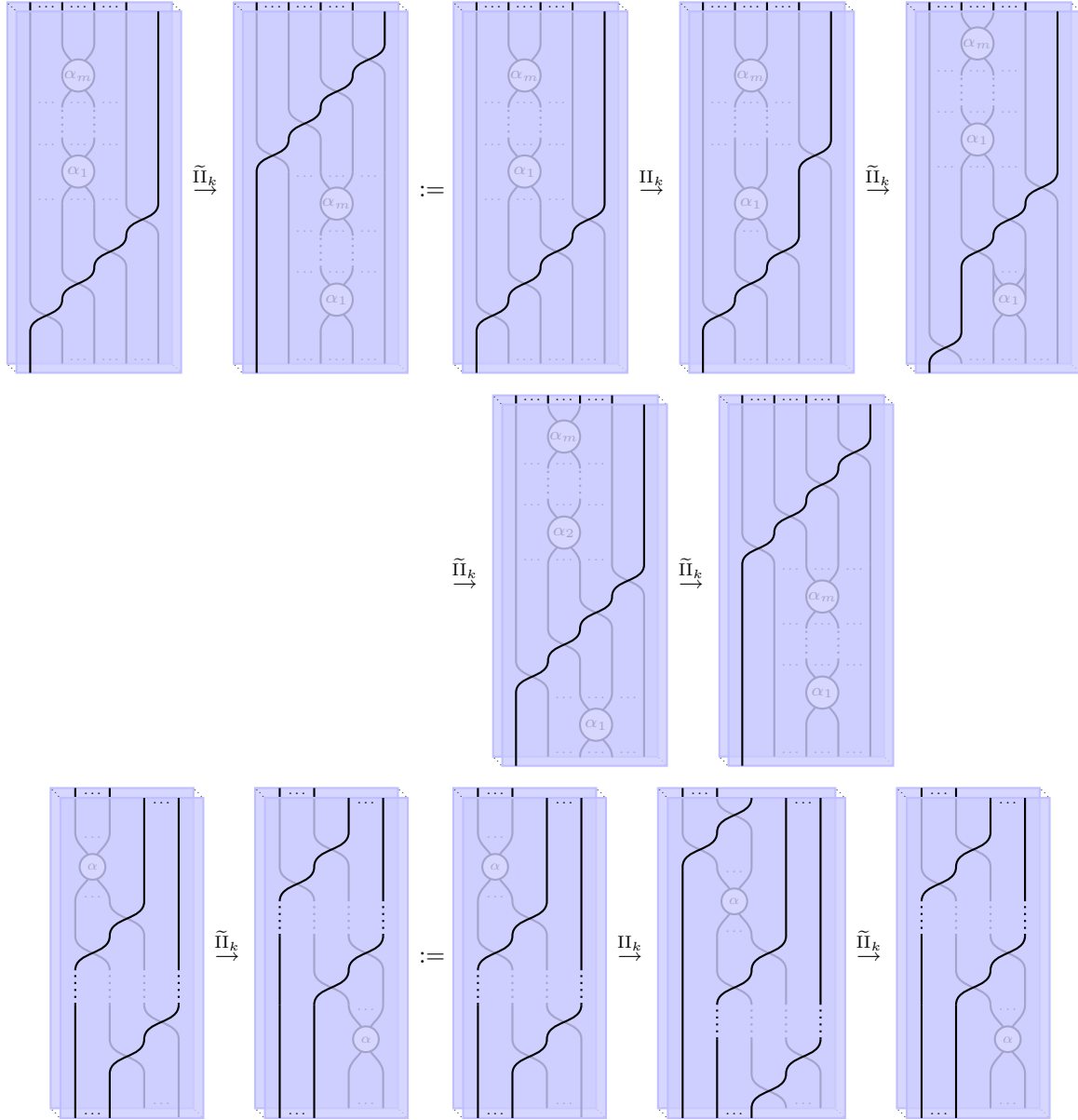


Fig. 7: Expansion scheme for type II homotopy generators.

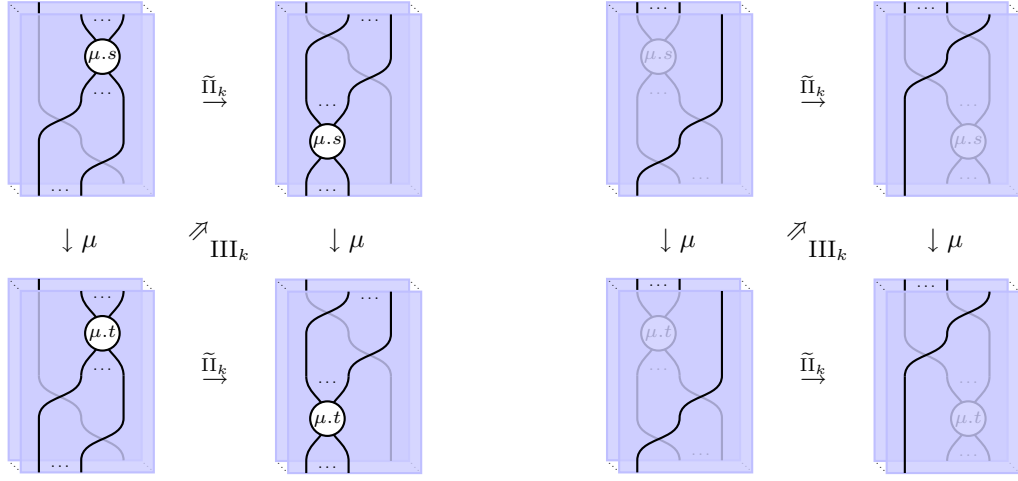


Fig. 8: The type III homotopy moves.

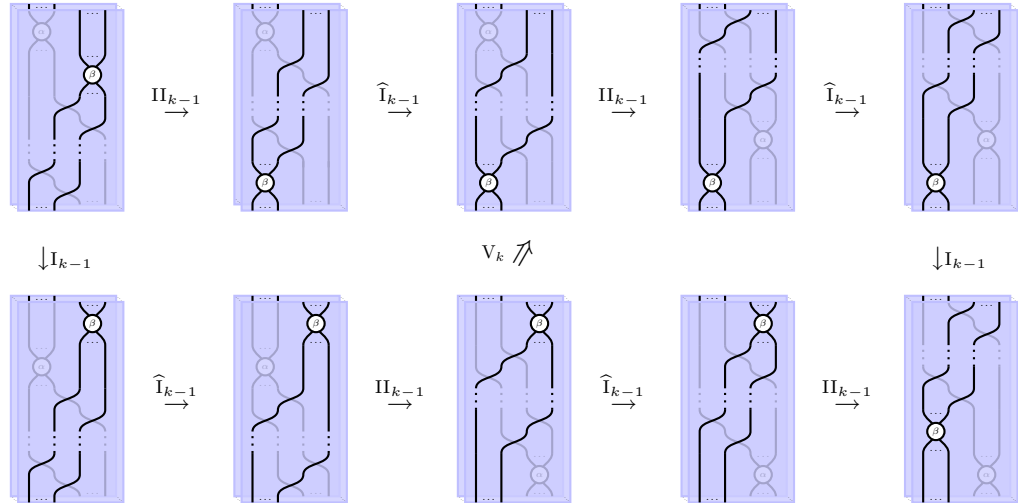


Fig. 9: A type V homotopy move.

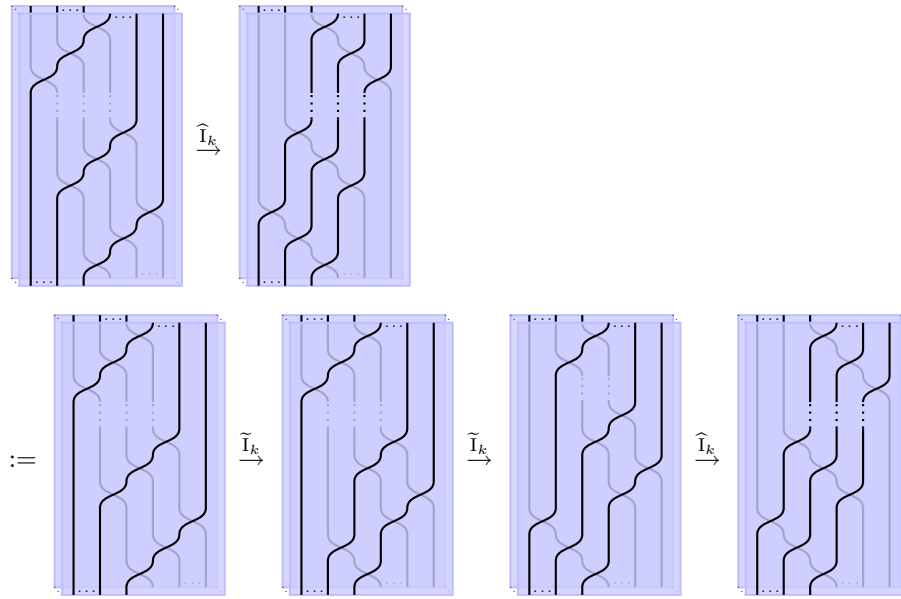


Fig. 10: The type I rearrangement scheme.

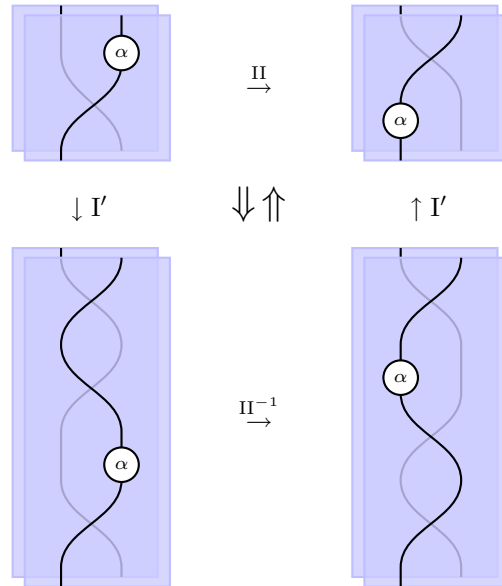


Fig. 11: Type VI homotopy moves.