Automated rewriting for higher categories and applications to quantum theory

Krzysztof Bar

University College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2016

Acknowledgements

First and foremost I would like to express the most immense gratitude to my supervisor Dr Jamie Vicary, whose patience and dedication always went far above and beyond the call of duty. As our weekly meetings often turned into daily discussions, where he taught me about the intricacies of category theory and quantum physics, but most importantly about how to express scientific results in a rigorous, yet accessible fashion. The guidance he provided gave me the sense of security, the importance of which cannot be understated in the life of a research student. Ever since we started working together in the middle of my first year, there has never been a slightest doubt in my mind that our research will lead to a successful completion of my DPhil. At the end of it all, I hope that I kept the promise I gave him during our first meeting, and he saved more time as the result of our collaboration, than he put into teaching me.

There are other distinguished academics who I had the fortune of encountering on my path. I want to thank my co-supervisors prof. Bob Coecke and prof. Samson Abramsky, who welcomed me as a member of their research group and initiated the entire research programme of categorical quantum mechanics, which has had such a profound impact on my academic life. My undergraduate college tutor prof. Andrew Ker for giving me a chance to come to Oxford all these years ago to begin the most transformative experience of my entire life. Prof. Artur Ekert, for making it possible for me spend the summer of 2014 at the Centre for Quantum Technologies in Singapore, which was one of the most scientifically prolific periods of my entire DPhil. Prof. John Baez whose guidance and encouragement during the initial stages of my work on higher category theory were invaluable. I would also like to thank Dr Sergusz Trzeciak for his mentorship and advice on career planning.

One's academic journey begins much earlier than at the moment they first set a foot in a university lecture theatre, for that reason I want to thank my high school and middle school mathematics teachers Boguslaw Kraszewski, Pawel Walter, Wioletta Kaszuba, for sparking my interest in the subject and teaching me that, in pursuing mathematics, there is no value more important than hard work.

I am thankful for the time spent with fellow doctoral students at the Quantum group at the Oxford University Computer Science Department. In particular I want to thank Linde Wester and Dominic Verdon, with whom I shared research interests and ideas on how to survive and thrive in the academic environment.

The completion of this thesis and my studies at Oxford in no way would have been possible without the support of a large number of people who I am proud to call my friends. All of them deserve much more than just a short recognition in this thesis, but I hope that this small token of appreciation can begin to repay them for their friendship and the innumerable acts of kindness.

Firstly, I want to thank Kuba Redlicki for the eight years spent together in Oxford both on and off the volleyball court and for showing me that both in sports and in life in general 'only the next point matters'. Joanna Bagniewska for her friendship and countless hours spent discussing plans on how to improve Poland, the United Kingdom and the world in general. I am not sure whether success here is realistic, but I am certain that we will give it our best shot. And to Robert Pisarczyk for all he has taught me about quantum physics and about science in general, especially during the summer we spent as flatmates in Singapore.

I would also like to acknowledge the many friends and collagues at the Oxford University Polish Society and the Federation of Polish Student Societies in the UK. Working and spending time alongside you sometimes felt almost like pursuing a parallel degree in public service. In particular I would like to acknowledge Jan Domanski, Ewelina Gregolinska, Aleksandra Janusz, Maciej Lisik, Mateusz Mazzini, Mikolaj Firlej, Anna Muszkiewicz, Aleksandra Pedraszewska, Marek Przytula, Bartosz Redlicki, Mateusz Ujma, Maria Wilczek and Andrzej Wolniewicz. I thoroughly enjoyed your company during the countless conferences, formal dinners and less formal parties afterwards and I know many more will come in the future. You provided me with welcomed distractions from the sometimes mundane reality of life as a research student and for that I sincerely thank you. Although I am certain that without your (always much appreciated) interference I would have finished this thesis many months ago.

Special thanks to Marta Tondera and Anna Bialas, who I had the pleasure of mentoring for a period of time, however in the end I am sure that I have learnt from you at least as much as you did from me. To Magdalena Richter and Tomasz Cebo for the shared interest in popularisation of science and the time we spent together on 'Science. Polish Perspectives' building a multi-disciplinary community of researchers. To my friends from the time as an undergraduate at University College, Oxford: Amy Bealey, Michael Captain, Dan Waters. Our reunion meetings in Oxford, London and Poole, as well as the many stimulating conversations always provided me with a much needed respite and allowed me to return to research with a rejuvenated interest. Also to those, especially Joanna Porebska and Przemyslaw Pobrotyn, who proofread the drafts of this thesis and provided many helpful remarks.

My years at Oxford would not have been complete without the friends and teammates at the Oxford University Volleyball Club: Michael D'Angeli, Rowan Hamill-McMahon, Alexander Stockenhuber and many others throughout the years. But the most heartfelt thanks go to the 2015/2016 team that won the BUCS Trophy, finished 3rd in the UK in the EVA student Cup and won the Varsity match: Fabio Anza, Andreas Iskra, Gytis Jankevicius, Sanders Lazier, Adam McBride, Christos Mavrokefalos, Stefan Nekovar, David Novotny, Jonas Pollex, Kuba Redlicki, Sven Sabas, Nicolas Stone-Villani and Tilo Zollitsch. The feeling of togetherness and the sense of comradery that we shared throughout the season is something that I will cherish for the rest of my life, especially on the days when I return to the Cape of Good Hope for a Wednesday drink to celebrate yet another victory.

I am forever indebted to my friends from home: Jakub Hyla, Michal Kulak, Kamil Morawski, Jan Porebski, Juliusz Sikora, Krzysztof Slupinski, Jan Tarnawski and Milosz Tatarczuch for all the time we spent together and for giving me something to look forward to in the future. But most importantly for, whenever I return to Krakow, making me feel as though I had never left.

Finally, I would like to thank my parents Anna and Waldemar, who both sacrificed so much to make it possible for me to pursue the most ambitious goals, I am deeply grateful for their unwavering support and for challenging me to always try to be the best version of myself. Thank you to my brother Piotr for his healthy dose of skepticism and that I can always count on him, also to my grandparents Emilia, Teresa, Bronislaw and Marian for always being there for me.

When I first begun my studies of cateory theory, I encountered the following quote by Stefan Banach:

> A mathematician is a person who can find analogies between theorems, a better mathematician is one who can see analogies between proofs and the best

> mathematician can notice analogies between theories. One can imagine that the ultimate mathematician is one who can see analogies between analogies.

Myself, I have often struggled to even understand the theorems and never aspired to be anything more than a mathematician. Nonetheless, the experience of being a research student, as deeply humbling as it may have been at times, is one which I will always hold dearest. Thank you, from the bottom of my heart, to all those who made it possible.

Abstract

This thesis consists of two contributions built on the foundation of higher category theory. The first is a novel framework for rewriting in higher categories. Its theoretical foundation is the theory of quasistrict higher categories and the practical realisation is a proof assistant Globular. Building on this, we propose a new definition of a quasistrict 4-category, and prove a result that in a quasistrict 4-category, an adjunction of 1-morphisms gives rise to a coherent adjunction satisfying the butterfly equations. The second contribution is the application of a higher categorical formalism to quantum theory to show equivalence between mutually unbiased bases and satisfaction of quantum key distribution schemes, and to prove correctness for construction of a particular set of solutions to the Mean King problem.

Contents

Chapter 1

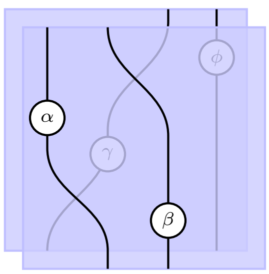
Introduction

Mathematics is the study of conceptual systems and processes that can be described with the rules of logic. To achieve a more thorough understanding of an abstract idea, one may decide to study ideas that are different but related, *i.e.* share certain similar properties. Focusing on the similarities between these ideas may yield further insights and lead to a deeper appreciation of the original concept. The study of relationships between mathematical structures is the domain of category theory, in the words of Eugenia Cheng [16] described as the *mathematics of mathematics*.

Going further, given some mathematical concepts and relationships between them, it may be of interest to study analogies between these relationships and going even further, to study analogies between analogies and so on *ad infinitum*. All with the purpose of gaining a better understanding of the original concept. The mathematical discipline that allows us to adapt this line of inquiry is called *higher category theory*. The adjective '*higher*' denotes the higher degrees of analogies that we study. The discipline has found a wide variety of applications, especially in homotopy theory, differential geometry and more recently in quantum theory, where the compositional nature of the theory plays a pivotal role.

Concentrating on abstract concepts is not necessarily in contrast with the study of physical processes, as mathematics provides an abstract model for what we observe in the real world. It depends on the particular context what level of abstraction is most desirable. For example, let us take the set of axioms for the real numbers. From the point of view of foundations of mathematics, it is of great interest to delve deep and analyse different models for these axioms, such as constructions from rational numbers using convergent Cauchy sequences or by Dedekind cuts. On the other hand, while performing everyday arithmetic calculations one does not need such a deep level of detail. Especially, if one is just interested in the result. In that case, keeping track of all the axioms used in the process would make the derivation much too cumbersome.

Similarly in category theory, there are instances when a higher level of abstraction gives additional insights. Various mathematical techniques may be applied to abstract away from the low-level details, and one such tool is graphical languages. For example, a composite 3-cell in a 3-category may be graphically presented as follows:



In this picture, each of the three ways in which two 3-cells can be composed together corresponds to a different spatial dimension: y-axis for 2-composition, x-axis for 1-composition and finally z-axis, perpendicular to the plane of the page, for 0-composition. Graphical languages for category theory work exceptionally well, precisely because they are easily readable for humans and play to the strengths of our cognitive abilities. At the same time, while they can be used as an intuitive notational shorthand, in some cases they can also be made entirely rigorous. In the literature, graphical calculi for n-categories have been formally developed for $n \leq 3$ [9, 29]. Completeness and correctness theorems for these calculi have been proved for monoidal categories [29] and for bicategories [55,61]. Building on this, we conjecture that similar results can be obtained for higher categories of dimension $n \geq 3$.

The crucial advantage of the graphical approach, in the spirit of finding the right level of abstraction, is the ability to absorb some of the axioms and make the category less burdensome to work with. Another benefit is the possibility to make the inherent geometrical nature of some algebraic proofs more explicit. A comprehensive survey of graphical theories for monoidal categories and weak 2-categories is due to Selinger [50].

Graphical languages are however not the only technique that simplifies reasoning about higher categories. The second method that we utilise in this thesis is the concept of simplifying some rules governing the behaviour of the mathematical object, without disrupting its equivalence to the original structure. An example of this is the notion of quasistrict n-categories, where one aims to strictify some of the equalities in weak n-categories while still maintaining equivalence to them. However, as the dimension and the complexity of categorical structures and their graphical representations increases, both methods of simplifying the reasoning about higher categories quickly exhaust their suitability. This is where the need for *automated reasoning* arises. In this thesis, we combine all three techniques to provide a framework for automated reasoning about quasistrict n-categories and their graphical representations and apply it to prove a result on coherent adjunctions in a quasistrict 4-category.

One discipline where category theory and graphical languages used to reason about it have found a particularly successful application is quantum computing. In the research programme of categorical quantum mechanics initiated by Abramsky and Coecke [1,2], a model of a quantum information processing system in the language of category theory limits the need to directly refer to the underlying Hilbert space. Instead, it allows us to concentrate on the topological flow of information within the system. Therefore, letting us find a useful level of abstraction with the aid of graphical languages. This is in contrast with reasoning about quantum computation in von Neumann's language of Hilbert spaces and linear maps, which has been likened by Abramsky and Coecke to trying to write a program on a classical computer using an assembly language of 0's and 1's.

1.1 Key contributions and thesis outline

The main contributions of this thesis are summarised in the list below. We present the following:

(1) New theory of generic-position higher-dimensional diagrams, with a combinatorial description.

(2) Framework for the definition of quasistrict higher categories, requiring considerably fewer axioms than traditional approaches.

(3) Application of this framework to propose a new definition of a quasistrict 4-category.

(4) Proof that an adjunction of 1-morphisms in a quasistrict 4-category gives rise to a coherent adjunction satisfying the butterfly equations, utilising the definition of a quasistrict 4-category. It is the first fully-detailed proof to our knowledge of a non-trivial result conducted explicitly in the setting of a 4-category.

(5) A completely syntactic proof within the higher categorical framework that a basis satisfies quantum key distribution if and only if it is mutually unbiased.

(6) A logical correctness proof of Klappenecker and Roettler's [33] construction of a solution to the Mean King problem from a family of mutually unbiased bases.

This thesis is organised in seven chapters. In this introductory chapter we provide a short primer on higher category theory, outlining its basic premises and giving intuition on the graphical calculi that can be used to reason within it. We give motivation for using the concept of *quasistrict* n-categories and introduce the notions of a *signature* and a *diagram*. These structures are used to develop a framework for automated reasoning about quasistrict n-categories. We also give an example of a proof in the graphical formalism by showing that in a strict 2-category every equivalence gives rise to an adjoint equivalence.

In Chapter 2, we introduce a new framework for automated reasoning and rewriting for quasistrict higher categories. As its theoretical foundation, we define two mutually-recursive structures. The first is a *signature* for a higher dimensional rewriting system, whose definition builds on the notion of an n-polygraph. Intuitively, a signature can be thought of as the collection of generating cells for a quasistrict n-category. The second structure is a generic-position higher-dimensional diagram which, in turn, is intended to correspond to the notion of a composite cell in a quasistrict n-category. We then discuss presentations of higher categories and how this notion could be leveraged to provide the definition of a quasistrict n-category using the signature structure. To the best of our knowledge such a combinatorial scheme for quasistrict higher categories has never previously been described. We define some basic operations on diagrams, such as *rewriting* and *composition* and conclude the chapter by proving a series of results on associativity and distributivity of diagram composition, which directly correspond to the axioms imposed in the traditional definitions of higher categories. This material also serves as correctness proofs for algorithms described in Chapter 4.

We further develop these structures in Chapter 3, where we endow signatures with the notion of a non-trivial *interchanger* morphism. In doing so, we follow Gray's approach [25] that involves strictifying all associator and unitor morphisms in a weak n-category and only leaving the interchanger morphisms non-trivial. We generalise the interchange law for 2-cells in a weak 2-category to the interchange of k-cells in a weak n-category and explore the plethora of higher-level coherences that arise from it. This allows us to give definitions of quasistrict 2- and 3-categories as signatures that support the interchange law and the appropriate higher-level coherences. We then proceed to show that a certain 4-signature satisfies the axioms of a switch 3-category [22], which is an alternative presentation of a **Gray**-category. As a result, we propose a new definition of a quasistrict 4-category as a 5-signature based on the same premise. In the final part of the chapter, we show how some standard categorical constructions could be retrieved in this setting, culminating in the proof that a triply degenerate 5-signature supporting certain higher-level coherences is a symmetric monoidal category.

In Chapter 4, we give a system description of the proof assistant Globular. The tool is based on the framework for reasoning about quasistrict n-categories constructed in Chapters 2 and 3. We discuss the technologies used and justify the design choices made. This is followed by an outline of the algorithms implemented and comments on their relation to the operation of diagram composition defined in Chapter 2. Finally, we give a short introduction to the user interface and discuss some possible extensions of the tool.

We use the new definition of a quasistrict 4-category to establish a result on adjunctions in a quasistrict 4-category in Chapter 5. We extend the result on strengthening an equivalence in a 2-category shown in Theorem 1.3.4 in this chapter, to prove that an adjunction of 1-morphisms in a quasistrict 4-category gives rise to a coherent adjunction satisfying the butterfly equations. Adjunction is a notion slightly weaker than an equivalence, where a 1-morphism is not invertible, but in some sense has a left inverse 'from below' and a right inverse 'from above'. Of special interest are coherent adjunctions, which carry even more structure. It is generally expected that an adjunction of 1-morphisms in a weak n-category gives rise to a more coherent adjunction. This has been proved for an arbitrary n by Verity and Riehl [47] assuming the homotopy hypothesis, which our result for a quasistrict 4-category does not depend on. The proof is conducted entirely in the graphical calculus and has been formalised with the aid of Globular giving further evidence for the correctness of the definition of a quasistrict 4-category. The main derivation consists of over 140 5-cell transformations and, if it was to be expressed as sequences of lower level cells, would consist of several thousand diagrams. For that reason it would be infeasible to carry out the proof without the assistance of automated reasoning. To best of our knowledge this is the first substantial proof of a non-trivial property conducted explicitly in the setting of a 4-category.

In Chapter 6, we discuss applications to quantum theory. We use symmetric monoidal 2-categories to formalise the notion of complementarity and describe the quantum key distribution protocol. We also prove correctness of construction of a particular set of solutions to the Mean King problem, building on the approach first proposed by Vicary [60]. The higher categorical formalism has never previously been used to describe quantum protocols of comparable complexity.

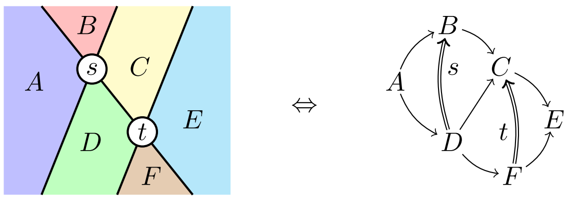
Finally, in Chapter 7, we discuss some areas of potential future investigation, including an extension of the definition of a quasistrict 4-category to $n \geq 5$ and a deeper analysis of the properties of the higher dimensional rewriting system defined by n-polygraphs.

1.2 Background on higher categories

Higher category theory is the study of *n-categories*. In addition to objects and morphisms present in ordinary category theory, an n-category is endowed with higher-level morphisms. In the same way as morphisms allow us to talk not only about equality, but also about isomorphism of objects in traditional category theory, a 2-morphism (or a 2-cell) allows us to talk about isomorphism of 1-morphisms (1-cells). In this setting objects are referred to as 0-cells. In addition to 1-cells between objects, an n-category has 2-cells which are morphisms between 1-cells, and in general $(k+1)$-cells between k-cells up to level n. There are n distinct composition operations that let us combine these into composite cells.

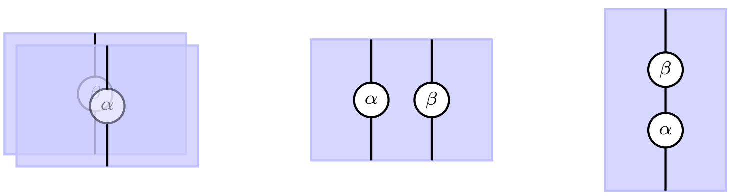
In the graphical calculus, a k-cell in an n-category is represented as an $(n-k)$-dimensional geometrical object. Due to this fact, the graphical representation of a k-cell is different depending on the n-category that we work in. This leads to the notions of string diagrams for 1-categories and planar diagrams for 2- and 3-categories.

For example, in a 2-category the standard way of graphically expressing 0-cells is as surfaces, 1-cells as wires and 2-cells as points. In the example below, we could see how a traditional pasting diagram denoting a composite 2-cell could be translated into a graphical representation:



The location of corresponding cells is the same in both pictures. 2-cells have been transformed into points, 1-cells into wires and 0-cells into regions. As stated earlier, vertical composition corresponds to placing 2-cells on top of each other along the y-axis and horizontal composition to placing them next to each other along the x-axis.

In a 3-category the dimensions of geometrical representations of cells increase by one, so that 0-cells are now represented by volumes, 1-cells by surfaces, 2-cells by wires and 3-cells by points, which we draw as little labelled circles to increase the clarity of presentation. Two 3-cells can now be composed in three distinct ways, which corresponds to glueing the diagrams together along one of the three boundaries:



In the case of 0-composition, we place the sheet containing the 3-cell α *in front* of the sheet containing β. This is realised by a quasi-3D picture and the effect of transparency of the front sheet. For 1-composition we place the sheets *next to* each other and for 2-composition *on top* of one another. Alternatively, a

3-cell can be represented by a pair of 2D pictures, which in this context can be thought of as a pair of snapshots of the source and the target boundary of the 3D representation, where the former transitions smoothly into the latter. Depending on the context we make use of either of these variants.

Remark on colour and transparency. The diagrams in this thesis make essential use of colour and transparency. We therefore recommend reading this document on a screen, or as a colour printout. For printing we recommend Adobe Reader, as some other PDF viewers do not correctly handle transparency.

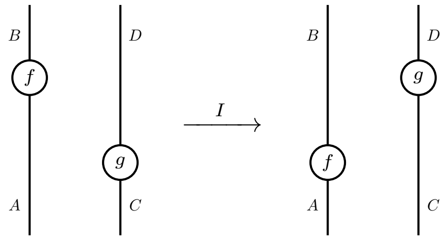
1.2.1 Quasistrict n-categories

Higher category theory studies relationships between objects, one such relationship is the concept of two objects (or at higher level relationships) being 'the same'. The axioms governing cell composition could either be more *strict* or more *weak* with regards to defining the notion of equality between cells. On one end of the spectrum we have weak n-categories, where rules such as associativity or unit law hold only up to higher-dimensional morphisms. This is in contrast to the other extreme of strict n-categories, where these rules are simple equalities. For instance, given a composite cell $\alpha \circ (\beta \circ \gamma)$, in a strict category associativity of composition is expressed by the equality: $\alpha \circ (\beta \circ \gamma) = (\alpha \circ \beta) \circ \gamma$. However, in a fully weak category one would only make a statement about these two composite cells being related by a higher-level associator morphism $\alpha \circ (\beta \circ \gamma) \to (\alpha \circ \beta) \circ \gamma$, which is a member of an invertible natural family of morphisms indexed by α, β, γ.

Weak n-categories are more prevalent in many disciplines of mathematics, however due to the presence of these additional families of cells, they can be more difficult to work with. Each family of associator, unitor or interchanger morphisms has to satisfy certain higher-level coherences, which themselves form higher-level invertible morphisms subject to coherences. This process leads to a fast growth of the list of axioms, so that simply stating all of them for n as low as 5 takes many pages and conducting any meaningful calculation would be unmanageable.

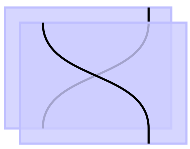
Strict n-categories are far easier to define [37], however their expressive power is limited. To illustrate this, let us consider the following *interchange* law for 2-cells in a weak 2-category expressed in the form of a 3-cell:

$\alpha = (f \circ_1 1_D) \circ_2 (1_A \circ_1 g) \quad :=$

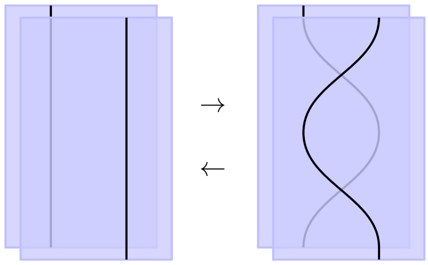


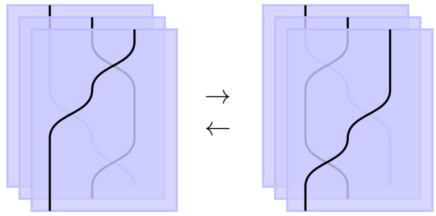
$:= (1_D \circ_1 g) \circ_2 (f \circ_1 1_C) = \beta$

When we consider a 3D graphical representation of this 3-cell, we see that it retrieves the familiar notion of a braiding:



This 3-cell behaves like a proper topological braiding and produces a cornucopia of higher-level coherences. Throughout the reminder of this thesis, we interchangeably refer to these either as singularities, coherences or interchangers. Some example coherences for $n = 4$ include:





When we consider the interchange law in a strict setting the two morphisms, α and β defined above, are instead strictly equal and there is no 3-cell between them, *i.e.* we have:

$$\alpha = (f \circ_1 1_D) \circ_2 (1_A \circ_1 g) = (1_D \circ_1 g) \circ_2 (f \circ_1 1_C) = \beta$$

In terms of a diagrammatical calculus this corresponds to two distinct graphical representations of *the same* morphism. This way, the braiding 3-cell and all the related higher-level morphisms do not arise. As a consequence, starting from $n = 3$, the notion of a strict n-category is no longer equivalent to the notion of a weak n-category [25]. It is then of limited use that strict n-categories have an elegant finite presentation using *n-polygraphs* [14] or *n-computads* [54] which was used to develop and realise in practice a higher-dimensional rewrite theory [42].

This presents a clear need for a notion that would maintain equivalence to a general weak n-category, but at the same time would be easier to work with. In this context, easiness is directly related to the length and the simplicity of proofs within the structure. One method to achieve that is to take a weak n-category and strictify as many of its rules as possible without disrupting the equivalence to a weak n-category. This is an informal description of what a *semistrict* n-category is. We take a slightly enhanced approach in which we are willing to accept a larger set of non-strict rules, as long as it leads to simpler proofs of optimal length. Semistrictness may, in general, be achieved in two distinct ways: one is to follow Gray's approach of turning the unitor and associator morphisms into identities and leaving the interchange law non-trivial. An alternative method, due to Simpson [52], keeps unitors non-trivial while making associators and interchangers strict. We follow the first of these paths and use the term *quasistrict* to refer to any n-category that has strict associator and unitor morphisms, but is still equivalent to a general weak n-category.

We compare this approach to other attempts to define semistrict n-categories. On the spectrum between fully weak n-categories and strict n-categories, the signature structures have a similar degree of strictness as the *switch* 3-*category* due to Douglas and Henriques [22], which is an alternative presentation of a **Gray**-category. In some aspects they are however slightly less strict than the semistrict 4-category, referred to as *4-teisi*, proposed by Crans [20]. The essence of these differences is that we do include some additional variants of higher-level coherences. This is however for a good reason, as it allows us to construct proofs with less bookkeeping and ultimately higher clarity.

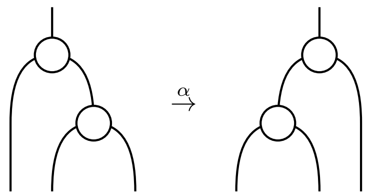
Semistrict and quasistrict n-categories for $n > 3$ are not, as of yet, well-understood. In fact even for $n = 4$, there are no widely accepted definitions. This is why, in Chapter 3, we propose a new definition of a quasistrict 4-category. In the reminder of this thesis we follow the practice of explicitly stating whether we are dealing with a weak, quasistrict, semistrict or strict n-category, however if in later chapters instances with no direct characterisation appear, assume that they are referring to a quasistrict n-category.

1.2.2 The need for automation

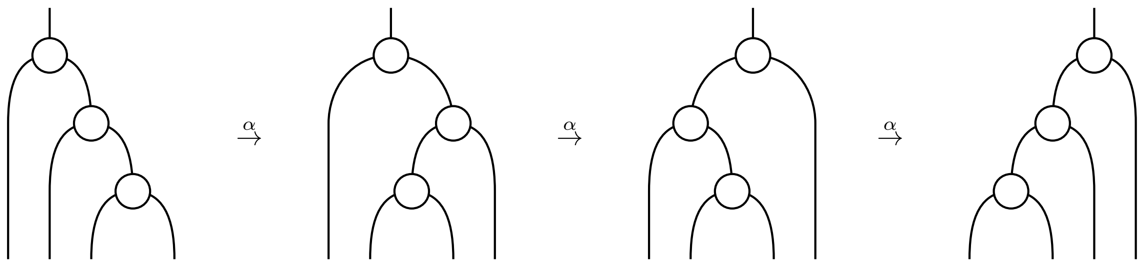
As mentioned before, with the rising values of n, higher n-categories become increasingly difficult to reason about, even with the aid of graphical calculus and the concept of quasistrict n-categories, giving rise to the need for automated reasoning. A proper framework for automation should have a firm mathematical foundation. Additionally, it should allow us to formalise a categorical proof in the graphical language, to reason about diagrammatical structures and to prove their properties directly without the need to refer to the underlying category. Kissinger created such a framework for symmetric monoidal categories [31]. As stated before, in this thesis we propose a framework for *quasistrict* n-categories.

We adapt an approach to higher category theory that emphasizes *rewriting*. For a k-cell $\alpha : f \to f'$ in a weak n-category, where f, f' are $(k-1)$-cells, we consider α to be a process that rewrites f into f'. The main benefit of this approach is that it provides a certain unity between the notions of *composition* and *proof*. A composite k-cell $\beta = \beta_m \circ \beta_{m-1} \circ ... \circ \beta_1$ such that $b_i : g_i \to g_{i+1}$ can be viewed as a proof that a $(k-1)$-cell g_1 can be rewritten to g_{m+1}. But in fact, every individual k-cell β_i could also be thought of as a *lemma* that g_i rewrites into g_{i+1}. Then, by composing all the separate k-cells β_i, we produce a multi-step proof of some more complex property of $(k-1)$-cells. For example, given a 3-cell α that witnesses an associativity rule of 2-cells:

α =



We can prove the following theorem about associativity, the proof presented as a composition of 3-cells:



A composite k-cell can be then *defined* as rewrite sequence between $(k-1)$-cells. In the context of this observation, for a quasistrict n-category, interchangers between n-cells may be realised as $(n+1)$-cells. Hence, in fact, the most convenient setting to work in is an $(n+1)$-category where these interchangers can be included as morphisms within the category instead of just being listed as standalone rules on top of an n-category.

1.2.3 Periodic table of higher categories

A useful feature in higher category theory is that trivialising the lowest-level cells in an n-category produces categories of lower dimensions that instead carry more *structure* for instance: braiding, syllepsis or symmetry [4]. The following table, as given in [5], presents some basic relations between categories: its columns are labelled by the dimension n of the category and its rows by the number m of levels where elements are stabilised. For instance, for $n = 4$, $m = 3$, we have a 4-category with 0-, 1-, and 2-cells trivialised which gives a symmetric monodial category.

	2	3	4	5	6
0	2Cat	3Cat	4Cat	5Cat	6Cat
1	MonCat	Mon2Cat	Mon3Cat	Mon4Cat	Mon5Cat
2	CommMon	BrMonCat	BrMon2Cat	BrMon3Cat	BrMon4Cat
3	-	CommMon	SymMonCat	SylMon2Cat	SylMon3Cat
4	-	-	CommMon	SymMonCat	SymMon2Cat

This table only involves weak n-categories, however we can still make use of it in our theoretical framework. There are three steps to this approach: Firstly, given a quasistrict n-category prove that it is equivalent to a semistrict n-category. Secondly, show that a semistrict n-category is equivalent to a weak n-category. Finally, use the periodic table of higher categories to transition into a weak category of a lower dimension that in exchange has more structure. At the moment, the highest dimension where equivalence between a weak n-category and a semistrict n-category has been proved is $n = 3$ [25]. As stated before, in Chapter 3 we propose a definition of a quasistrict 3-category, based on the signature structure, which is equivalent to a **Gray**-category 3.5.2. Based on that, for $n = 3$ we have the following process:

- Gray 3-category is equivalent to a weak 3-category
- Weak 3-category with one object is a weak monoidal 2-category
- Weak monoidal 2-category with one object is a braided monoidal category

Hence, by modelling a Gray 3-category, we are also able to model a braided monoidal category. For $n = 4$ this method would yield a symmetric monoidal category, however to make this claim completely rigorous, a coherence theorem for quasistrict 4-categories would need to be proved.

An important disclaimer at this stage is that currently there are no coherence theorems for semistrict or quasistrict n-categories for $n \geq 4$. Moreover, as stated before, there are not even any widely accepted definitions of semistrict or quasistrict n-categories for these values of n. It is therefore entirely possible, that there may exist some weak n-categories which are not equivalent to any quasistrict n-category and, as such, cannot be modeled in our approach. It also remains to be shown for $n = 4$ that the definition of a quasistrict n-category based on the signature structure that we propose in Chapter 3 gives a weak n-category.

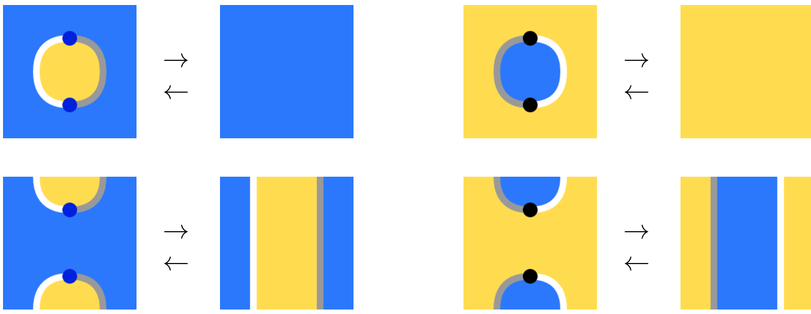
1.3 Proofs in the graphical calculus

The notion of equivalence between two mathematical structures is often captured by an isomorphism between them. In higher category theory this may however be not sufficient, since both invertible cells witnessing the isomorphism are themselves subject to higher-level morphisms. For example, for an isomorphism witnessed by $A \xrightarrow{f} B$ and its inverse, we have morphisms $f \circ f^{-1} \xrightarrow{\alpha} \mathrm{id}_A$ and $\mathrm{id}_B \xrightarrow{\beta} f^{-1} \circ f$. In a 2-category this is expressed formally as follows:

Definition 1.3.1. In a 2-category, an *equivalence* is a pair of objects A and B, a pair of 1-cells $A \xrightarrow{F} B$ and $B \xrightarrow{G} A$ and invertible 2-cells $F \circ G \xrightarrow{\alpha} \mathrm{id}_A$ and $\mathrm{id}_B \xrightarrow{\beta} G \circ F$, denoted as follows:

$\alpha \quad =$  $\beta \quad =$ 

Invertibility of α and β is captured by the following 3-cells:



Note that equations between 2-cells may be expressed in the form of invertible 3-cells that act as rewrites between both sides of the equality. Here, these 3-cells are depicted in the form of 2D snapshots of the source and the target. A special case is where the 2-category is **Cat**, in which case this yields the usual notion of equivalence of categories.

In a weak n-category, for A and B to be equivalent, we also require α and β to be each other's inverses and similarly for all the higher-level equivalences up to n. If all these morphisms are invertible, we say that A and B are equivalent in a maximally weak sense. This is a desirable property, as equivalent objects can be replaced by one another in any investigation without affecting the outcome, therefore truly allowing us to gain insight about a mathematical object by studying another related structure.

In practice however, there are instances where maximally weak equivalences do not arise. In those cases, we may instead study less strong notions that nonetheless give us some insight about the related objects. One such notion is adjunction, where a 1-morphism which is a part of it, may be regarded as having a left inverse 'from below' and a right inverse 'from above'. This is formally expressed in a 2-category by the following:

Definition 1.3.2. In a 2-category, an *adjunction* is a pair of objects A and B, a pair of 1-cells $A \xrightarrow{F} B$ and $B \xrightarrow{G} A$ and a pair of 2-cells $F \circ G \xrightarrow{\alpha} \mathrm{id}_A$ and $\mathrm{id}_A \xrightarrow{\beta} G \circ F$, denoted as follows:

α =  β = 

That satisfy the following *snake equations*:

 =   = 

Many interesting features of category theory, such as universal constructions or limits, arise from the presence of adjunctions.

On the other hand, equivalence is not the most structured manner in which two objects could be related. The notions of equivalence and adjunction are not incompatible and we may combine them to produce a more coherent notion of equivalence:

Definition 1.3.3. In a 2-category, an equivalence is called an *adjoint equivalence* if it additionally satisfies the snake equations, as described in Definition 1.3.2.

In 3- and 4-categories in a similar manner, more coherent notions of adjunctions could be obtained. In Chapter 5, we show two results of this type. Firstly, that in a 3-category an adjunction of 1-morphisms gives rise to a more coherent adjunction satisfying the swallowtail equations. Secondly, that in a quasistrict 4-category an adjunction of 1-morphisms gives rise to a more coherent adjunction satisfying the butterfly equations. In particular this second result is, to the best of our knowledge, the first substantial proof in the setting of a 4-category given in the literature. As the result shown was generally expected in the community, this constitutes even further evidence for the correctness of the definition of a quasistrict 4-category.

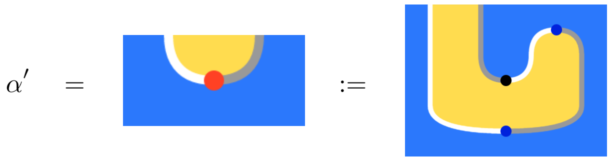
To give the reader a flavour of what proofs in the graphical calculus are like, we present an example where a well-known fact that an equivalence in a 2-category gives rise to an adjoint equivalence [36] is proved. Since we are operating in a 2-category, a 0-cell is graphically represented by a surface, a 1-cell by a line and a 2-cell by a point.

Theorem 1.3.4. *In a 2-category, every equivalence gives rise to an adjoint equivalence.*

Proof. Given an equivalence witnessed by α, β, by Definition 1.3.3, we need to build an equivalence that additionally satisfied the snake equations 1.3.3. Let α and β be invertible 2-cells that are graphically represented follows:

α =  β = 

Then, let us consider the following 2-cell α' constructed from α^{-1}, β and α :

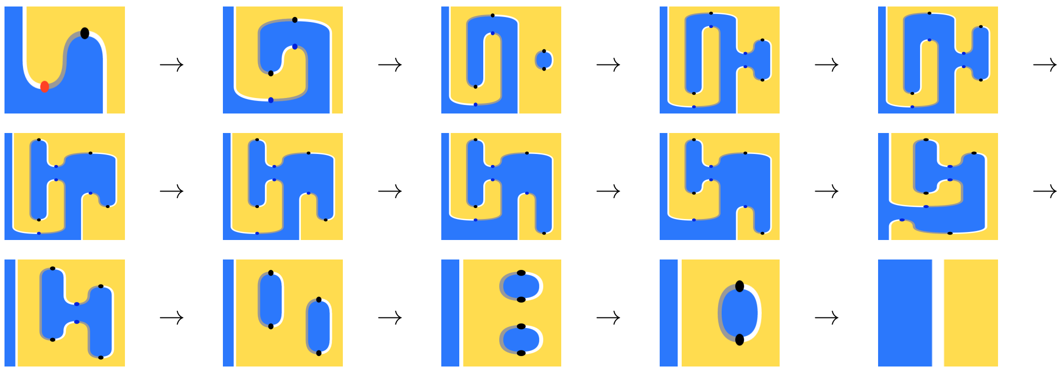


We want to show that α' and β witness an adjoint equivalence, for that we need to satisfy two conditions:

- α' and β are invertible, hence witness an equivalence
- α' and β satisfy the snake equations

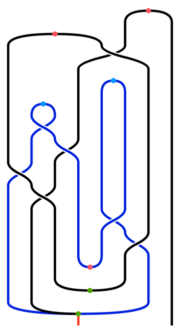
Now, in line with the rewriting perspective, to prove that an equation is satisfied means to show that the left hand side could be rewritten into the right hand side by a series of (3-cell) rewrites. First, since α and β are invertible, invertibility conditions outlined in Definition 1.3.1 hold.

Since both α and β are invertible, invertibility of α' straightforwardly follows. To show the satisfaction of the first snake equation, we perform the following sequence of rewrites:

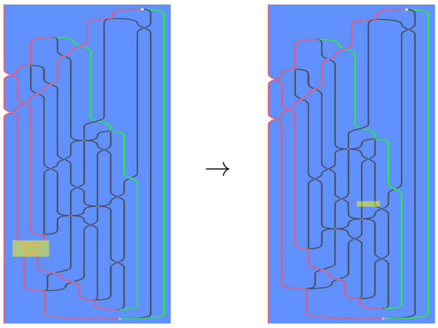


The other snake equation follows in a similar manner, hence all conditions are satisfied and α' and β witness an adjoint equivalence, as required. □

The sequence of rewrites showing that the first snake equation is satisfied forms a composite 3-cell. We may gain some additional insight about the topological nature of the proof by viewing it in its entirety as a single diagram:



The graphical representation of this 3-cell is expressed here in a 2-dimensional view with the lowest dimension projected out, so we can look at this entire 3-cell 'side-on'. The nodes represent applications of rewrite rules, and the wires represent 2-cells. From this view, we can examine the proof and possibly simplify it by eliminating redundant steps (e.g. a rewrite immediately followed by its inverse) or by re-arranging rewrites that are applied to independent parts of the diagram. In Chapter 5, we use similar techniques to show an extension of this result for 4-categories: that an adjunction of 1-morphisms gives rise to a coherent adjunction satisfying the butterfly equations. An example 5-cell step in that derivation is expressed in the graphical calculus by the following two pictures:



To explain the meaning of these two pictures and of the composite 3-cell representing the proof of Theorem 1.3.4, firstly we discuss the details of the graphical formalism in Section 3.1. Secondly, we describe how non-trivial interchanger morphisms in quasistrict 3- and 4-categories can be illustrated in this notation in Sections 3.2.2, 3.2.3. Finally, in Section 4.1 we demonstrate how to use projections into $\mathbb{R}^2$ for graphical representations of 4- and 5-cells and, as a consequence, how to render them using Globular. As stated earlier in the introduction, the main derivation consists of 140 steps of this kind, if we were to express them all without the use of projections a couple thousand pictures would be needed. The necessity to keep track of all this data clearly demonstrates the need to employ techniques of automated reasoning to conduct proofs in n-categories for dimensions as low as $n = 4$.

1.3.1 Globular

On the basis of the theory presented in this thesis we created a proof assistant Globular. It is a practical realisation of the techniques for automated reasoning about higher categories that we develop in Chapter 2 which implements the theory of quasistrict categories, as presented in Chapter 3. Globular is an online tool for formalisation and verification of higher categorical proofs that allows users to hyperlink proofs directly into their research papers. It produces graphical visualisations of higher-dimensional proofs and type checks to prevent malformed cells from being constructed. It adapts the perspective of higher-dimensional rewriting, *i.e.* proofs being viewed as sequences of rewrites between graphical representations of lower level objects.

As explained earlier in this chapter, as the dimension of categorical structures increases, graphical languages alone become no longer sufficient to make the arising complexity manageable. This creates the need for a framework automated reasoning. The final step and the ultimate test of usefulness for such a framework is practical implementation.

Globular is built on the foundation of two basic structures: signatures and diagrams, as defined in Chapter 2. Due to their mutual dependence, the user may build both structures simultaneously. Generators in the signature could be used to create increasingly complex diagrams, which in turn could be designated as sources and targets of higher level cells added to the signature. Additionally, Globular supports two methods of modifying diagrams. The first is composition, as abstractly defined in Definition 2.4.6, which corresponds to the categorical notion of composition. The second is rewriting, as abstractly defined in Definition 2.3.4, which, in accordance with the higher dimensional rewriting perspective, allows us to create proofs.

The current capability of the tool is modelling quasistrict categories, as defined in Chapter 3, up to and including dimension $n = 4$. Even though there exist other tools for automated reasoning about symmetric monoidal categories [32] and opetopic higher categories [24], none of them have capabilities comparable to Globular, which is the first tool of its kind.

The overarching goal with Globular is to create a proof assistant that would be intuitive to use and genuinely useful for the category theory community. In the 9 months since deployment, the tool, which is hosted on the web at `http://globular.science`, has been used over 8000 times by 1800 unique users, who uploaded 36 proof formalisations into the publicly available gallery on the Globular webpage. We hope that as we further enhance the tool's capabilities these numbers will increase even further.

1.4 Signatures and Diagrams

In Chapter 2 we introduce a pair of mutually-recursive diagram and signature structures that serve as the basis of the theoretical model for automated reasoning in *quasistrict* n-categories. Here we give an informal introduction to these structures and give an example that exhibits the essence of an approach that emphasizes quasistrictness. Intuitively, a signature can be thought of as the set of generating cells for an n-category and a diagram as a specific composite k-morphism built from the elements in the signature.

Producing a general scheme for automated reasoning in quasistrict n-categories poses significant difficulties. Firstly, graph-like features of diagrams used in Kissinger's formalism for symmetric monoidal categories, as described in [31], are no longer present in higher dimensions. Secondly, an arbitrary composite n-cell can be used as the source or target of an $(n+1)$-cell, which is itself a building block of a composite $(n+1)$-cell, so a structure that allows that has to be employed. Finally, as dimensions go up, for $n \geq 4$ it quickly becomes difficult to visualise the structures that are being modeled. Because of all these reasons, generic-position higher-dimensional diagrams for quasistrict n-categories have never previously been defined and their definition is an original contribution of this thesis.

The key innovation is the total order structure imposed on n-cells in an n-dimensional diagram. Recall that, in a weak n-category the same cells composed in different orders could give different composite cells as the outcome. These in turn need to be related by higher level morphisms, which for quasistrict n-categories are non-trivial interchangers. An example of this phenomenon is given for horizontal composition in a 2-category in Subsection 1.2.1, where this gives rise to the interchange law.

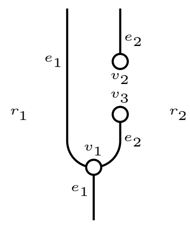
Diagrams endowed with the order inducing data capture the substance of the quasistrict approach and the implicit order that it induces on the elements of a composite n-cell. Without the order inducing data, one would instead model the axioms of a strict n-category. To illustrate the importance of this concept, we present examples of diagrams D_i over the same signature σ that consist of the same generators but differ in the order they are composed in. In a strict setting, all these diagrams would be equal, however in a quasistrict approach we need to distinguish between them and subsequently relate them by higher-level cells: *interchangers*. Labels in these diagrams are to be thought of as types of the individual components. Here, we informally use the graphical notation for the diagram and signature structures for illustrative purposes. The notation is formally specified in Definition 3.1.1, once correctness of the abstract definitions of diagrams and signatures is proved in Chapter 2.

Let σ be as follows:

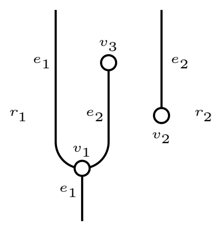
$$\sigma = \{\{r_1, r_2\}, \{e_1, e_2\}, \{v_1, v_2, v_3, v_4, v_5\}\}$$

Then, the different individual diagrams and the combinatorial data representing them are as follows:

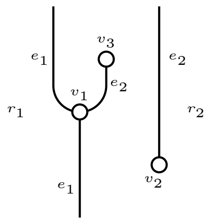
$$D_1.g = [v_1, v_3, v_2]$$
$$D_1.e = [[0], [1], [1]]$$
$$D_1.s = [(e_1, [])]$$



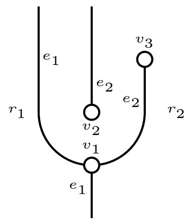
$$D_2.g = [v_1, v_2, v_3]$$
$$D_2.e = [[0], [2], [1]]$$
$$D_2.s = [(e_1, [])]$$



$D_3.g = [v_2, v_1, v_3]$
$D_3.e = [[1], [0], [1]]$
$D_3.s = [(e_1, [])]$



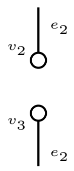
$D_4.g = [v_1, v_2, v_3]$
$D_4.e = [[0], [1], [2]]$
$D_4.s = [(e_1, [])]$

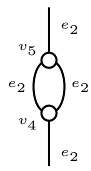


These 2-diagrams consist of the same components and differ only in the order in which they have been composed. This is reflected in the different numerical values within the lists that specify the position of a node in a horizontal slice. A comprehensive description of the graphical notation for combinatorial diagrams is given in Chapter 3.

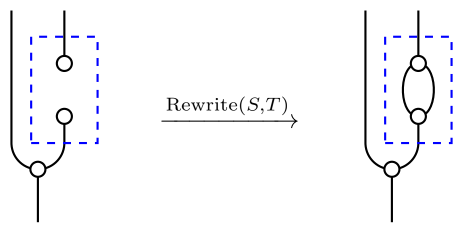
This has a profound effect on the operations of diagram composition and rewriting, whose abstract definitions we provide in Chapter 2. Here we give an intuitive overview on when these two procedures are applicable and what are the effects of applying them to a diagram. We use the example diagram structures defined above in order to further emphasise the impact of the order inducing data. In the presentation below, whenever the types of individual cells within diagrams are unambiguous, we omit the superficial type labels.

Intuitively, to 'rewrite' a diagram D means to replace some part of it S with another another diagram T, such that the resulting structure is still a valid diagram. There are certain criteria we could impose on S and T to ensure that. These are globularity conditions, whose name is inspired by the corresponding notion in higher category theory, *i.e.* we require sources and targets of S and T to match. We define this formally in Chapter 2, but informally this ensures that T could be inserted into the empty slot left by the removal of S from D, so all cells on the boundaries match. Let us consider the following example rewrite defined by S and T:

$S \quad =$ 

$T \quad =$ 

Let us then apply it to diagram D_1 first. We can see that S appears as a subdiagram in D_1, replacing S with T results in the rewritten diagram, the location of both S and T is marked with blue dashed rectangles, which are not a part of the graphical formalism:

$D_1 \quad =$ 

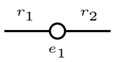
Now note the diagram S does not appear as a part of the remaining diagrams D_2, D_3 and D_4. This is for the following reasons:

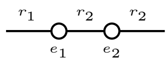
- In D_2, the node v_2 appears before v_3 in the list of generators $D.g$. Additionally, a relation between edges is introduced, so that the incoming edge of v_3 is to the left of the outgoing edge of v_2.
- In D_3, the node v_2 appears before both v_1 and v_3 in the list of generators $D.g$
- In D_4, similarly as for D_1, the node v_2 appears before v_3 in the list of generators $D.g$. Additionally, a relation between edges is introduced, so that the incoming edge of v_3 is to the right of the outgoing edge of v_2.

Note how in D_3, because of the ordering of v_3 and v_2, neither of their edges is to the left or right to the other. As S does not arise within D_2, D_3 and D_4, then the rewrite cannot be applied. The only diagram rewritable using S and T is D_1.

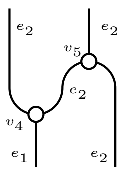
The second diagram modifying operation is composition, which is formally defined in Chapter 2. Intuitively, this procedure allows us to 'glue' together two diagrams that share a common boundary. These do not necessarily have to be of the same dimension. In this context, a *source boundary* of an n-diagram is to be understood as the $(n-1)$-diagram created by only considering the $(n-1)$-cells that are not in the target of any of the n-cells in the diagram. The intuition for the *target boundary* is analogous. We again illustrate this with the aid of the previously defined diagrams D_1, D_2, D_3, D_4.

Note that, even though the ordering of nodes within each of these diagrams is different, this does not affect the cells in either the source or the target boundary. Hence for all four example diagrams we have the same source boundary B_s and the same target boundary B_t, which are both 1-diagrams:

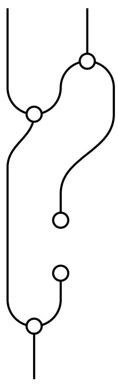
B_s = 

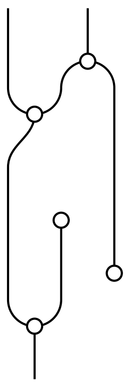
B_t = 

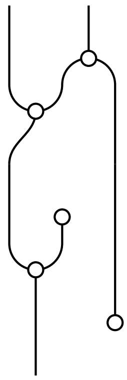
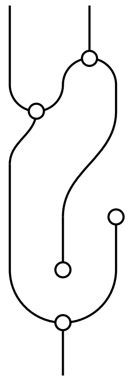
Note that the 1-cells in the graphical representations of the 1-diagrams above are expressed as 0-dimensional geometrical objects and 0-cells as 1-dimensional geometrical objects. Now, given an example diagram C:

C = 

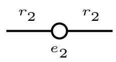
The vertical composites of all four diagrams D_1, D_2, D_3, D_4 with the example 2-diagram C exist, as they share a common boundary. These are as follows:

$D_1 \circ C$ = 

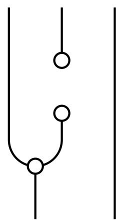
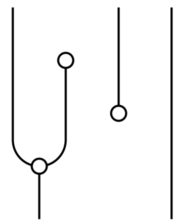
$D_2 \circ C$ = 

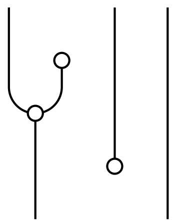
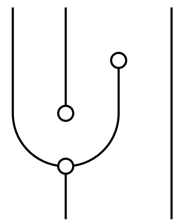
$D_3 \circ C \quad =$  $D_4 \circ C \quad =$ 

We can also compose each of the example diagrams with a 1-diagram C'. This is because the target boundaries of B_s and B_t, which have to be the same due to globularity conditions, match the source boundary of C' (all are equal to r_2):

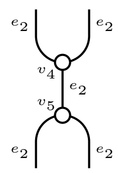
$C' \quad =$ 

Then, the composites of D_1, D_2, D_3, D_4 with C' are as follows:

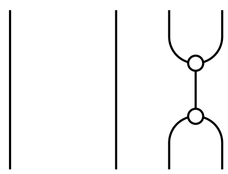
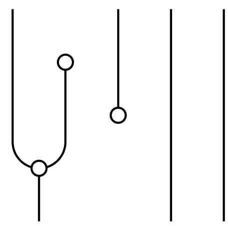
$D_1 \circ C' \quad =$  $D_2 \circ C' \quad =$ 

$D_3 \circ C' \quad =$  $D_4 \circ C \quad =$ 

We conclude this section with an observation that an n-diagram D and an m-diagram S can be composed in exactly one way, along their matching $\max(m,n) - \min(m,n) + 1$ boundary. For two n-diagrams this corresponds to the categorical notion of vertical composition. All other methods of composition present in higher categories are retrieved via the use of whiskering. This is exemplified by horizontal composition of D_1 with the following diagram C'':

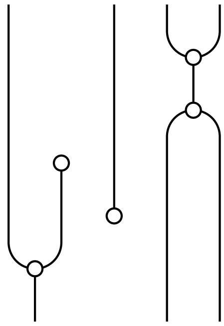
$C'' \quad =$ 

Here, we compose the target boundary of D_1 with C'' and D_1 with the source boundary of C'':

$t(D_1) \circ C'' \quad =$  $D_1 \circ s(C'') \quad =$ 

Then, we vertically compose the resulting diagrams:

$$(D_1 \circ s(C'')) \circ (t(D_1) \circ C'') \quad =$$



An important note is that, equivalently, we could have chosen to compose D_1 with the target boundary of C'' and the source boundary of D_1 with C'' to create the composite $(C'' \circ s(D_1)) \circ (t(C')' \circ D_1$. This is precisely the subject of the interchange law that gives rise to diagrams that need to be related by higher level interchanger morphisms which we explore further in Chapter 3.

Chapter 2

Automated rewriting for higher categories

In this chapter, we introduce a new framework for automated reasoning and rewriting for quasistrict higher categories. First we discuss the existing methods, which to the best of our knowledge are only applicable to strict n-categories. We then define the notion of a signature for a higher dimensional rewriting system, formalise the notion of a diagram over this signature and use it to solve the problems of rewriting and composition. We prove correctness of both constructions in Sections 2.3 and 2.4. Finally, we show some results on associativity and distributivity of diagram composition in Section 2.5. These results are later used to concisely express the axioms on associativity and distributivity of cell composition when we use the signature structure to define quasistrict 2-, 3- and 4-categories in Chapter 3. As all the notions involved are mutually dependent, the proof technique used for all these results is to give logical statements that depend on the dimension n of the structure and then inductively proving that the conjunction of all the statements holds for all $n \geq 0$.

An important question is in the motivation for developing all these new intricate structures to give a definition of a quasistrict 4-category, while a more standard framework of listing all the axioms could be used. The main advantage of the method used in this thesis is that, even though the proofs of correctness of these constructions as well as distributivity and associativity of composition are complicated, the actual definitions of quasistrict higher categories are vastly simpler than in traditional approaches. For instance the definition of a switch 3-category due to Douglas and Henriques [22] consists of 34 axioms plus additional variants for reflexions and rotations, while in the definition of a quasistrict 3-category given in Chapter 3 all distributivity and associativity results are summarised by two Theorems about structures that are specified in this chapter using only a few definitions.

The crucial advantage however, lies in the scalability of our approach. For $n \geq 4$ we do not need to say anything more on associativity and distributivity of composition, as the results proved in this chapter hold for all $n \geq 0$ with no additional complexity. At the same time, a definition in a traditional style would need to list many additional axioms explicitly. With regards to singularities, we list them in a systematic manner with complexity comparable to that in the traditional approach. An additional benefit of a definition using the diagram and signature structures is its particular suitability for the purposes of automation. We use this fact in Chapter 4 where we discuss a practical proof assistant Globular that was built on the basis of this approach.

In introducing the background material in this chapter, we follow the exposition by Mimram [42]. In particular we concentrate on the notions of n-computads [54] and n-polygraphs [14] that are built on the foundation of globular sets. We describe their structure and strictness properties and contrast this with our approach. We also discuss the double pushout rewriting (DPO) formalism for graph rewriting [31].

The key original insight in this chapter is the explicit order structure imposed on the list of cells within a diagram. This lets us capture the implicit ordering of individual k-cells within a composite k-cell which is a characteristic feature of a quasistrict n-category that has non-trivial interchangers. Throughout, we informally use the graphical notation for the diagram and signature structures for illustrative purposes. This is to give graphical intuition as to the function of the auxiliary notions that we define. To formally specify the graphical notation, which we do in the next chapter in Definition 3.1.1, we need to first prove correctness of the operations of diagram rewriting and composition. This is shown further in this chapter, by proving Theorems 2.3.27 and 2.4.19.

2.1 Rewriting systems and presentations of n-categories

Intuitively, a rewrite of a mathematical structure is the result of replacing some part of this structure with new elements, under the requirement that after the replacement, the structure is still sound. Examples in different areas of mathematics range from deductive rules in logic to graph rewriting. Once a rewriting system has been defined, various properties such as confluence or termination of rewriting paths can be investigated. There is a well-established notion of a (one-dimensional) rewriting system, as defined in [38]:

Definition 2.1.1. A *rewriting system* is a pair of elements (A, R) such that A is a set and $R \subseteq A \times A$ is a binary relation on the set A.

The relation R provides us with the information on the rules of term replacement in a formal expression over A. Given our focus on higher category theory, we are however more interested in higher dimensional rewriting systems. In such a system, relations (rewrites) between its elements can be combined to form rewriting *paths*. These paths are then themselves subject to relations that rewrite one path into another. Similarly as for higher categories, the number of times we repeat this process gives us the dimension of the rewriting system.

The main difficulty with modelling such a system lies in the fact that, at each level, individual relations can be combined to form composite relations. In particular, we have to not only systematically and formally generate all the possible rewriting paths α_i between elements A_j in the system, but also generate relations Γ_k between the composites of these rewriting paths *etc.* An algebraic structure describing a higher dimensional rewriting system needs to generate collections of all such elements for all dimensions of relations up to n. This concept is formally captured by the notion of an n-polygraph. It was first introduced by Burroni [14] and later independently investigated by Street and Power [54] under the name of n-*computads*. Higher dimensional rewriting systems are also closely connected to the notion of a *presentation* of a category.

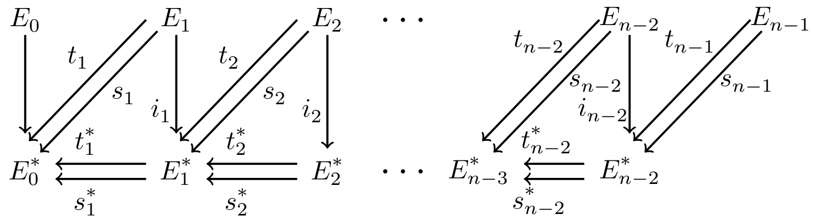
In algebra, there is a well-known concept of a presentation of a group, where by giving a list of generating elements and rules that the elements are required to satisfy we provide sufficient information to uniquely define the group. Presentation of a category is the extension of this concept. For 1-categories, we give a list of generating 0-cells and 1-cells, so that every object and morphism in the category can be expressed as a composite of these generators. Additionally, we provide a list of rules that the generating morphisms are supposed to satisfy, which can be expressed in the form of 2-cells. For $n \geq 2$ listing the generators and rules becomes more challenging, as a k-cell could have an arbitrary multidimensional composite $(k-1)$-cell as its source or target.

An n-polygraph is a structure that describes the generating cells for strict n-categories. The concept is a generalisation of the manner in which a directed graph describes the generating objects and generating morphisms of a 1-category, which together with a set of generating rules forms a presentation of a category. Polygraphs can be thought of as the most general structure generating a free n-category. Since the rules imposed on n-cells can be expressed as $(n+1)$-cells, an $(n+1)$-polygraph is a presentation of an n-category.

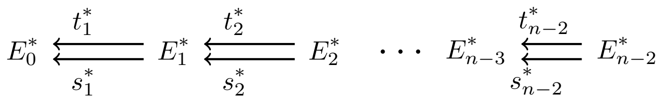
Definition 2.1.2. Given an integer $n \geq 0$ an *n-dimensional rewriting system* or an *n-polygraph* is defined as follows:

- For $n = 0$, a 0-polygraph is a set E_0.

- For $n > 0$, given an $(n-1)$-polygraph *i.e.* the following structure in **Set**:

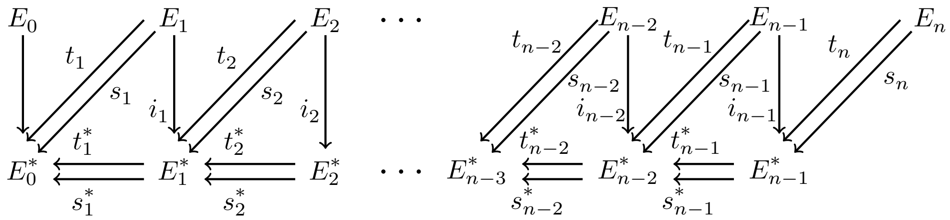


Together with a structure of an $(n-2)$-category on the $(n-2)$-graph:



Such that, this $(n-1)$-polygraph generates a free $(n-1)$-category, which we denote as E^*, with the above $(n-2)$-category as the underlying $(n-2)$-category and containing the elements of E_{n-1} as generators and the source and target given by functions s_{n-1}, t_{n-1}. Its set of composite $(n-1)$-cells, which are obtained by composing the generating elements in E_{n-1}, is denoted by E^*_{n-1}, the canonical injection by $i_{n-1} : E_{n-1} \to E^*_{n-1}$. Finally, the maps s^*_{n-1}, t^*_{n-1} denote the source and target maps for the composite $(n-1)$-cells in E^*_{n-1}. The meaning of all maps with lower indices is explained similarly using the recursive nature of this definition.

Given all that, an n-polygraph is defined as a structure of the following form in **Set**:



Together with the structure of an $(n-1)$-category on the graph at the bottom of the picture, such that:

$$s^*{}_{n-2} \circ s_{n-1} = s^*{}_{n-2} \circ t_{n-1} \qquad t^*{}_{n-2} \circ s_{n-1} = t^*{}_{n-2} \circ t_{n-1}$$

Note that $E^*{}_0$ is the free 0-category on E_0. Since E_0 is a set, so is $E^*{}_0$ and the two are isomorphic. For an n-polygraph, we call the elements of each set E_k, k-generators. The n-category $\mathcal{E}$ presented by an $(n+1)$-polygraph E is obtained by quotienting the underlying n-category E^* by the relation relating n-cells α, β whenever there is an $(n+1)$-cell $\Psi : \alpha \to \beta$ in E^*. We say that a category $\mathcal{C}$ is presented by an $(n+1)$-polygraph E when $\mathcal{C}$ is isomorphic to $\mathcal{E}$.

2.2 Basic structure

In rewriting theory, signatures are algebraic structures containing building blocks from which rewriting terms can be built. We generalise from monoidal signatures to a setting similar to n-polygraphs, the difference remaining in how we define *diagrams.* The notion of a diagram corresponds to the notion of composite k-cells created from the set of generators E_k. Our choice of terminology is influenced by the graphical calculus and the diagrammatical manner in which these composite k-cells are ultimately expressed.

2.2.1 Signatures and diagrams

In the following exposition, due to the mutually-recursive nature of diagrams and signatures, there are many interdependent concepts and no clear-cut order of precedence in which they should be presented. For that reason, whenever a need arises to refer to a notion that has not been defined yet, we flag this explicitly, provide an intuition for the notion and refer the reader to the formal definition which is provided further in the section.

We begin by defining the two basic structures that our theoretical model is built on: *signatures* and *diagrams*. Their definitions are mutually dependent, however dimensions in the definitions decrease with each successive reference. This is because an n-signature only refers to $(n-1)$-diagrams, which in turn only refers to an $(n-1)$-signature. That way the ladder of mutual references terminates with a 0-signature and both structures are well-defined. We also define the auxiliary notion of a *diagram embedding* $e : S \hookrightarrow D$ to allow us to reason about one diagram being a *subdiagram* of another, this is formally stated in Definition 2.2.7.

Given a diagram D, there are two important ways by which this diagram could be modified:

- A subdiagram S embedded in a diagram D by an embedding e could be replaced by another diagram T to form a *rewritten diagram* $D.\Pi[e, T]$
- Two diagrams can also be *composed* along a common boundary to form a composite diagram

Since diagrams are intended to model composite cells, they are endowed with a source and target structure, such that the source of a diagram D, referred to as $s(D)$ and the target, referred to as $t(D)$ both are $(n-1)$-diagrams. We define both of these formally in Definition 2.2.5, once the definitions of a signature and of a diagram are in place. In some instances when we want to put emphasis on the source structure as an attribute of D, we use the notation $D.s$ to refer to it.

Definition 2.2.1. An *n-signature* σ is a collection of sets of cells $(G_0, ..., G_n)$, such that for all $g \in G_k$ and all $0 < k \leq n$, there are $(k-1)$-diagrams over $(G_0, ..., G_{k-1})$ we call $s_k(g)$, $t_k(g)$, such that $s(s_k(g)) = s(t_k(g))$ and $t(s_k(g)) = t(t_k(g))$.

Note that G_0 is the set of 0-cells that have no source and no target.

Definition 2.2.2. An *n-diagram* D over an n-signature $(G_0, ..., G_n)$ is a list of length $|D|$ such that each element $D[i]$ consists of the following:

- *Cell*: $D[i].g \in G_n$
- *Embedding* into the i-th *slice* of D: $D[i].e : s(D[i].g) \hookrightarrow D[i].d$, if this slice exists

and if $n > 0$, then the diagram also consists of:

- *Source*: $D.s$ which is an $(n-1)$-dimensional diagram over $(G_0, ..., G_{n-1})$

For an n-diagram D, its source $D.s$ can be rewritten by generators $D[i].g$ to obtain $|D|$ diagrams of dimension $n-1$, which we refer to as *slices*.

Definition 2.2.3. In an n-diagram D, its i-th *slice* $D[i].d$ is:

- For $i = 0$ we have: $D[0].d = D.s$
- For $i > 0$ we have: $D[i+1].d = (D[i].d).\Pi[D[i].e, t(D[i].g)]$

Note that, for $n = 0$, there is no source $D.s$, hence there are no slices, no embeddings and there is just a single cell $D[0].g$. Given an n-signature σ, we denote the set of k-diagrams over this signature as ${\Delta^*}_k(\sigma)$ or simply ${\Delta^*}_k$ if there is no ambiguity with regards to σ. For any generator $g \in G_k$ in σ there is a corresponding k-diagram, this is summarised by the following inclusion maps:

Definition 2.2.4. Given an n-signature $\sigma = (G_0, ..., G_n)$ there are *inclusion maps* $i_k : G_k \to {\Delta^*}_k$ such that for $g \in G_k$ we have:

$$i_k(g) = \delta$$

Here δ is a k-diagram such that $\delta.s = s_k(g)$, $|\delta| = 1$, $\delta[0].g = g$, $\delta[0].e = \mathrm{id}_{s_k(g)}$.

Now we can formally define the maps that relate an n-diagram D to its *source* and *target* $(n-1)$-diagrams.

Definition 2.2.5. For an n-diagram D there are two maps ${s_n}^*, {t_n}^* : {\Delta^*}_k \to {\Delta^*}_{k-1}$, that send a k-diagram D to its *initial* slice $D[0].d$ (source) or its *terminal* slice $D[|D|].d$ (target). If there is no ambiguity with regards to n, we simply refer to the maps as s, t.

These maps satisfy the usual globularity conditions $s(s(D)) = s(t(D))$, $t(s(D)) = t(t(D))$. In total, a diagram D has $|D| + 1$ slices, because there is one for the source of each of $|D|$ cells and an additional one for the target of the final cell.

We write $s^k(D)$ and $t^k(D)$ to denote taking the source or the target of the diagram k times, we also call these diagrams k-boundaries of D. When we want to refer to the individual lists instead of the overall diagram, we use $D.g$ for the list of cells, $D.e$ for the list of embeddings, and $D.l$ for the list of slices. For an n-diagram D, we say that it is of *dimension n*.

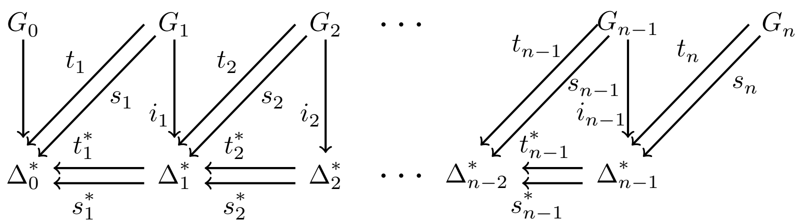
We formalise the concept of a diagram structure over a signature σ being sound or *well-defined* in the following way:

Definition 2.2.6. We say that an n-diagram D is *well-defined*,

- If $n = 0$: no conditions
- If $n > 0$: 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.

The main purpose of the subsequent sections in this chapter is to formally prove that both the procedure of rewriting and the procedure of composition preserve the property of being well-defined and thus could be used as the foundation for modelling the cell composition in higher categories.

The mutual dependencies between diagram and signature structures are summarised below. Given a signature $\sigma = (G_0, ..., G_n)$ the mutual dependences between diagrams and signatures can be expressed diagramatically:



This picture is the same as for n-polygraphs, however, the structure is more sophisticated, with diagrams containing information on the specific order in which cells in the diagram appear.

To consider some non-trivial modifications of diagrams, we first need to define the notions of:

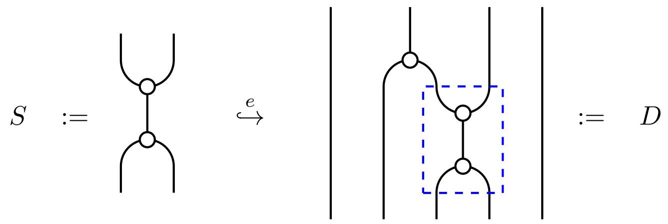
- *diagram embedding*: $e : S \hookrightarrow D$, Definition in Definition 2.2.7
- *lifted embedding*: $e.\Lambda[T] : T \hookrightarrow A.\Pi[e, T]$, described in Definition 2.3.1
- *embedding composition*: $e \circ f : S \hookrightarrow D \hookrightarrow A$, described in Definition 2.3.2

Note that an n-diagram only makes reference to embeddings of $(n-1)$-diagrams, so the dependencies are strictly descending.

Definition 2.2.7. Given two n-diagrams S and D, an *n-diagram embedding* $e : S \hookrightarrow D$ is given recursively as follows:

- $n = 0$: No data
- $n > 0$:
  - ⋆ *Height*: $e.h \in \mathbb{N}$
  - ⋆ *Component embedding:* $e.e : S.s \hookrightarrow D[e.h].d$

Intuitively, a diagram embedding $e : S \hookrightarrow D$ allows us to specify where the n-diagram S appears within the n-diagram D. This can be illustrated by the following example:



Here the numerical values in e are as follows: $e.h = 0$, $e.e.h = 2$. This is indicated in the picture by the presence of the dashed rectangle, which is superficial and not part of the graphical formalism. Note how the location of S in D is zero vertices from the bottom and two edges from the left in the 0-th slice.

Similarly as for the diagram structure, we need a formal way of saying that the given map is a proper embedding relating the corresponding cells and embeddings of both diagrams.

Definition 2.2.8. Given an n-diagram embedding $e : S \hookrightarrow D$ between well-defined diagrams S and D, we say that it is *well-defined* if its data satisfies the following properties:

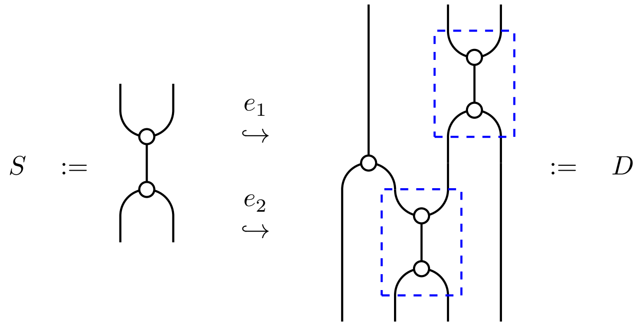
- If $n = 0$: we need $S[0].g = D[0].g$
- If $n > 0$:
  - ⋆ The component embedding $e.e$ is well-defined
  - ⋆ For every $0 \leq i < |S|$:

$$S[i].g = D[i + e.h].g \tag{2.1}$$

  - ⋆ For every $0 \leq i < |S|$:

$$(e.e).\Lambda[S[i].d] \circ S[i].e = D[i + e.h].e \tag{2.2}$$

An embedding of 0-diagrams is characterised by no data because 0-diagrams only consist of a single cell. In the course of this chapter we use this definition to show that each embedding we define is well-defined. These results are later used in proving that various diagram modifications are well-defined. Given two n-diagrams S, D, if there exists a diagram embedding $e : S \hookrightarrow D$, we say that S is a *subdiagram* of D witnessed by the embedding e. Note that, in particular, there may be more than one embedding $e : S \hookrightarrow D$. In this case, there are multiple instances of S being a subdiagram of D. This is illustrated by the following example:



Even though e_1 and e_2 both have the same domain and codomain, they differ in the numerical data that specifies the location of S within D: $e_1.h = 0$ and $e_2.h = 2$.

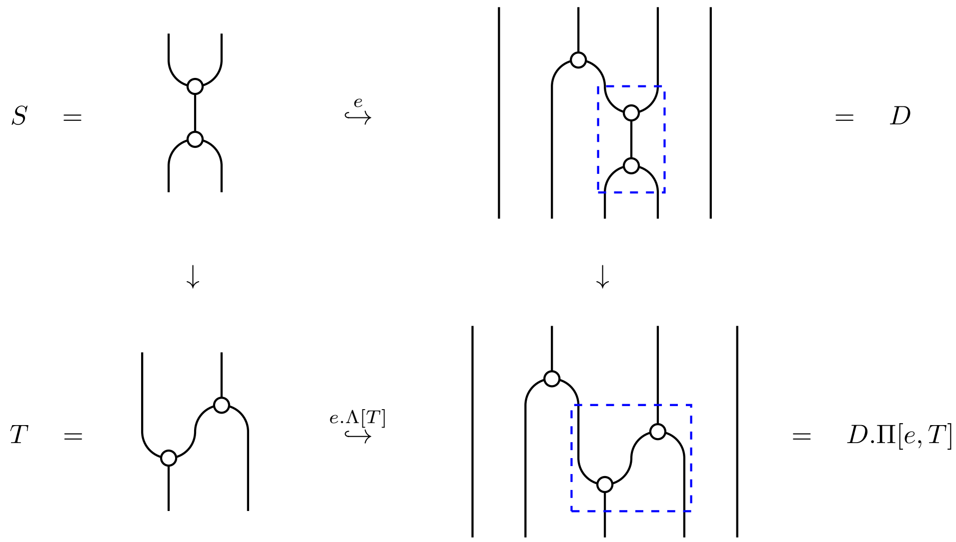
2.3 Rewriting

The intuition behind the procedure of rewriting is that we want to transform a diagram D, by removing a part of the diagram S embedded in D by the embedding $e : S \hookrightarrow D$ and replace it with another diagram T, somehow preserving connectivity between corresponding elements. For that reason we need $s(S) = s(T)$, $t(S) = t(T)$ and we need to update the relevant embeddings $T[i].e$ corresponding to the generators $T[i].g$ being added to D. To that end, we use the auxiliary notion of a lifted embedding $f.\Lambda[T]$, which allows us to construct an embedding of T in the rewritten diagram.

Definition 2.3.1. Let S, T, A all be well-defined n-diagrams such that $s(S) = s(T)$, $t(S) = t(T)$, let $e : S \hookrightarrow A$ be a well-defined embedding. Then, the *lifted embedding* $e.\Lambda[T] : T \hookrightarrow A.\Pi[f, T]$ is defined as follows:

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

We illustrate this with the following example:



Given an embedding $e : S \hookrightarrow D$, we 'lift' it to obtain an embedding of T in $D.\Pi[e, T]$. Both embeddings are indicated by the blue dashed rectangles.

The lifted embedding allows us to construct the embeddings in the rewritten diagram. We do this by extracting the data of a component embedding $e.e$ and use it to give an embedding of the i-th slice of T into the $(i + e.h)$-th slice of the rewrite of D.

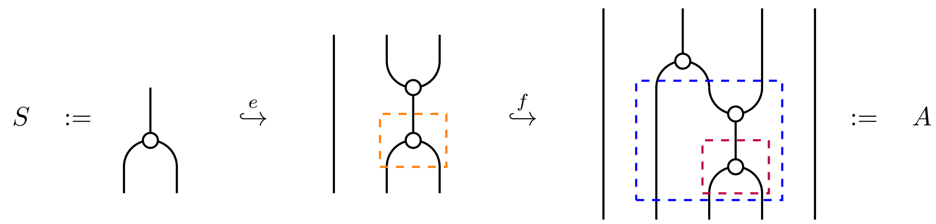
We also use the lifted embedding to define composition of two embeddings $e : S \hookrightarrow D$ and $f : D \hookrightarrow A$. This is necessary, since there is a mismatch between the source of $f.e$ and the target of $e.e$, and $(f.e).\Lambda[D[e.h].d]$ is needed to make the transition between them.

Definition 2.3.2. Given two n-diagram embeddings $e : S \hookrightarrow D$ and $f : D \hookrightarrow A$ their *composite* $f \circ e : S \hookrightarrow A$ is also a diagram embedding defined as follows:

- For $n = 0$, we define $f \circ e$ to have no data
- For $n > 0$, we define:

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

Intuitively this means that the relation of being a subdiagram of another diagram is transitive. Consider the following example chain of embeddings:



The diagram S is directly embedded in D by e, which is indicated by the orange rectangle. D in turn is embedded in A by f, indicated by the blue rectangle. These two can be combined together to obtain the composed embedding $f \circ e$ so that S is directly embedded in A, this is indicated in the picture by the purple rectangle.

The final auxiliary concept that simplifies reasoning about diagram modifications is globularity of diagrams:

Definition 2.3.3. Two n-diagrams S, T are *globular* with respect to each other if:

- For $n = 0$, no requirements
- For $n > 0$, we need $s(S) = s(T)$ and $t(S) = t(T)$

With all the auxiliary structures in place, we are now in the position to formally define the first modification of a diagram: a *rewrite*. Notice how for an n-diagram such that $n > 0$, the length $|D|$ of the list of generators and embeddings changes to $|D| - |S| + |T|$ as we remove $|S|$ elements of the source of the rewrite and replace them with $|T|$ elements of the target. For this reason the lists in the rewritten diagram consist of three segments: the initial and the final segment that remain unaltered and the middle slice which gets replaced. Depending on the value of $e.h$ and on $|S|$ any of the three segments in the rewritten diagram may be empty.

Definition 2.3.4 (Rewrite)**.** Given an n-diagram D with a subdiagram $e : S \hookrightarrow D$, and an n-diagram T globular with respect to S, the *rewrite* $D.\Pi[e, T]$ of D is the following n-diagram:

- If $n = 0$, then:

$$|D.\Pi[e, T]| = |D| - |S| + |T| \tag{2.3}$$
$$(D.\Pi[e, T])[0].g = T[0].g \tag{2.4}$$

- If $n > 0$, then:

$$(D.\Pi[e,T]).s = D.s \tag{2.5}$$

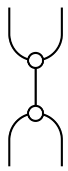
$$|D.\Pi[e,T]| = |D| - |S| + |T| \tag{2.6}$$

$$(D.\Pi[e,T])[i].g = \begin{cases} D[i].g & \text{if } 0 \leq i \leq e.h \\ T[i - e.h].g & \text{if } e.h < i < e.h + |T| \\ D[i - |T| + |S|].g & \text{if } e.h + |T| \leq i < |D| - |S| + |T| \end{cases} \tag{2.7}$$

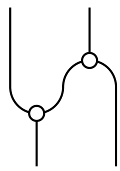
$$(D.\Pi[e,T])[i].e = \begin{cases} D[i].e & \text{if } 0 \leq i \leq e.h \\ (e.e).\Lambda[T[i - e.h].d] \circ T[i - e.h].e & \text{if } e.h < i < e.h + |T| \\ D[i - |T| + |S|].e & \text{if } e.h + |T| \leq i < |D| - |S| + |T| \end{cases} \tag{2.8}$$

This definition can be illustrated with an example which is of the same style as those given in Section 1.4. Consider the following diagrams S, T, D.

S =

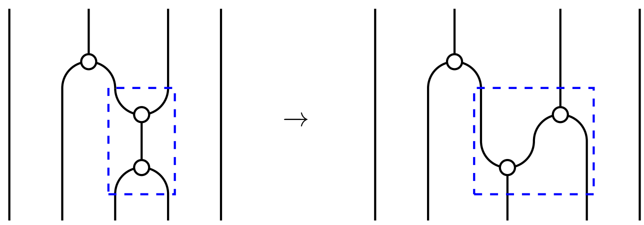


T =



Note that there is an embedding $e : S \hookrightarrow D$ denoted by the blue dashed rectangle. Additionally S, T are globular with respect to each other as their respective sources and targets match.

D =



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

In the resulting diagram $D.\Pi[e,T]$, the generators of S have been removed, and generators of T inserted instead. Their positions have been determined by the corresponding component embeddings. In this particular example the final segment is empty, as $e.h + |S| = |D|$.

Let us now compare this definition of a rewrite to the notion of double pushout rewriting (DPO), which is a method of formalising graph rewriting [31]. In the category **Graph**, a rewrite rule is defined by inclusions of boundaries of the source and the target of the rewrite into the graph being rewritten, while matches are represented by injective graph homomorphisms. This allows us to detect a match of the rewrite source, remove it and then glue the target of the rewrite in its place along the common boundary to produce the rewritten graph. Even though the graphical representations of our diagrams do have certain graph-like features for $n = 1, 2$, for larger n this is no longer the case. The most significant obstacle to employing the DPO method is the fact that a diagram D from which the source of the rewrite has been removed is no longer a well-defined diagram, as the rewrite rules for slices no longer hold. This is in contrast to graphs, where a graph with some part of its interior removed is still a graph. On the other hand, there are some DPO features exhibited by our structures, for example rewrite boundaries are matched against the rewritten diagram, the interior of the source of the rewrite is removed and the interior of the target of the rewrite is inserted, using boundaries to record the location of the match. But altogether, the DPO paradigm for rewriting is not applicable to diagram structures.

Note that attempting to rewrite a diagram D using an identity rewrite, results in D remaining unaltered. In this derivation, as well as in all those that follow in the reminder of this thesis, for each equality we always refer explicitly to the statement that justifies it. If the reference is absent and we don't specifically state upfront that all the equalities follow by a single result, then the particular equality follows by arithmetic manipulations. The following argument also gives a flavour of the proof style that is employed in the reminder of this chapter.

Lemma 2.3.5. *Given an n-diagram D with a subdiagram S, witnessed by an embedding $e : S \hookrightarrow D$, the following equality holds:*

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

Proof. By Definition 2.3.7, we need to check the following four conditions:

- Sources are equivalent diagrams: All equalities in this derivation follow by the source clause Eq. (2.5) in the definition of a rewrite.

$$D.\Pi[e, S].s = D.s \qquad \text{[Eq. (2.5)]}$$

- Sizes of lists of generators are equal

$$\begin{aligned} &|D.\Pi[e, S]| \\ &\quad = |D| - |S| + |S| && \text{[Eq. (2.6)]} \\ &\quad = |D| \end{aligned}$$

- Corresponding generators are equal, for $0 \leq j \leq |D|$. We consider this separately for three ranges:

  ⋆ For $0 \leq j \leq e.h$

$$D.\Pi[e, S][j].g = D[j].g \qquad \text{[Eq. (2.7)]}$$

  ⋆ For $e.h \leq j \leq e.h + |S|$

$$\begin{aligned} &D.\Pi[e, S][j].g \\ &\quad = S[j - e.h].g && \text{[Eq. (2.7)]} \\ &\quad = D[(j - e.h) + e.h].g && \text{[Eq. (2.1)]} \\ &\quad = D[j].g \end{aligned}$$

  ⋆ For $e.h + |S| \leq j \leq |D|$

$$\begin{aligned} &D.\Pi[e, S][j].g \\ &\quad = D[j - |S| + |S|].g && \text{[Eq. (2.7)]} \\ &\quad = D[j].g \end{aligned}$$

- Corresponding embeddings are equivalent, for $0 \leq j \leq |D|$. We consider this separately for three ranges:

  ⋆ For $0 \leq j \leq e.h$

$$D.\Pi[e, S][j].e = D[j].e \qquad \text{[Eq. (2.8)]}$$

  ⋆ For $e.h \leq j \leq e.h + |S|$

$$\begin{aligned} &D.\Pi[e, S][j].e \\ &\quad = (e.e).\Lambda[S[j - e.h].d] \circ S[j - e.h].e && \text{[Eq. (2.8)]} \\ &\quad = D[(j - e.h) + e.h].e && \text{[Eq. (2.2)]} \\ &\quad = D[j].e \end{aligned}$$

⋆ For $e.h + |S| \leq j \leq |D|$

$$\begin{aligned} &D.\Pi[e, S][j].e \\ &\quad = D[j - |S| + |S|].e && \text{[Eq. (2.8)]} \\ &\quad = D[j].e \end{aligned}$$

As all these four conditions are satisfied, the two diagrams $D.\Pi[e, S]$ and D are indeed equivalent. □

Before proceeding to prove correctness of the rewrite construction, we need to introduce the notions of and *diagram equivalence* and *embedding equivalence i.e.* a formal way to express that two diagrams and two embeddings are *the same*.

Definition 2.3.6. Two n-diagram embeddings $e : A \hookrightarrow B$ and $f : C \hookrightarrow D$ are *equivalent*, written $e = f$, if the following is satisfied:

- $A = C$ and $B = D$
- If $n > 0$: additionally both $e.h = f.h$ and $e.e = f.e$ hold

For two 0-diagram embeddings there is no data, hence it is sufficient if they have the same source and the same target to conclude that they are equivalent.

Definition 2.3.7. Two n-diagrams D and S are *equivalent*, written $D = S$, if the following is satisfied:

- $|S| = |D|$
- 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 $n > 0$, we also require $D.s = S.s$

Note that, for $n = 0$, there is just a single cell $D[0].g$ and no source and no embeddings that could be compared. Alternatively we could say that two diagrams S and D are equivalent if there exist two embeddings $e_1 : S \hookrightarrow D$ and $e_2 : D \hookrightarrow S$ such that $e_1 \circ e_2 = \text{id}_D$ and $e_2 \circ e_1 = \text{id}_S$. Then, the equivalence $S = D$ is *witnessed* by the pair of embeddings e_1, e_2. Throughout the reminder of this thesis, whenever we place a "=" sign between diagrams or embeddings, it is always to be understood in terms of Definitions 2.3.6 and 2.3.7.

2.3.1 Correctness of the rewriting construction

In the following subsection, we prove that the defined diagram modification - rewrite, forms a well-defined diagram. We also show that the lifted embedding and embedding composition used to construct the rewrite are well-defined. Due to the heavily recursive structure of our definitions, we introduce several logical statements: $P(n), R(n), S(n), T(n), Q(n), A(n), B(n), C(n)$ that are dependent on the diagram dimension n. All these statements concern properties of n-diagrams and embeddings between n-diagrams.

Of particular importance is the statement $R(n)$, which formalises the fact that a rewrite of a well-defined n-diagram is also well-defined. $B(n)$ and $C(n)$ state respectively that a lifted embedding between well-defined n-diagrams is well-defined and that a composite of two well-defined embeddings between n-diagrams is also well-defined. The remaining statements $P(n), S(n), T(n), Q(n), A(n)$ are auxiliary statements that are necessary to carry out an inductive proof that $R(n)$ holds for all $n \geq 0$, *i.e.* that the process of rewriting preserves the diagram property of being well-defined for any diagram.

The intuition behind the poof is that in order to say that a rewritten n-diagram $D.\Pi[e, T]$ is well-defined ($R(n)$), we need all its slices, which are $(n-1)$-diagrams, to be well-defined. To achieve that, we express them as slices of D or as rewrites of a well-defined $(n-1)$-diagram $D[e.h].d$ ($S(n)$, $T(n)$) and use the inductive hypothesis $R(n-1)$. For the inductive hypothesis to be applicable, the component embeddings $D.\Pi[e,T][j].e : s(D.\Pi[e,T][j].g) \hookrightarrow D.\Pi[e,T][j].d$, which are embeddings of $(n-1)$-diagrams, must be well-defined. In the segment of $D.\Pi[e,T]$ between $e.h$ and $e.h+|T|$, these embeddings are built using lifted embeddings (well-defined by $B(n-1)$) and embedding composition (well-defined by $C(n-1)$), as per Definition 2.3.4. However, for all these to be well-defined the main condition is well-definedness of rewrites of $(n-1)$-diagrams, so we are back to the original statement, but this time with the dimension n reduced by one.

After n iterations, we only need well-definedness of rewriting for 0-diagrams which follows with no further conditions and forms the base case of the proof. This way, we use the network of mutual dependencies and the well-foundedness of these structures to show the main result that rewriting preserves well-definedness of a diagram.

The remaining statements $P(n), Q(n)$ express certain properties of lifted embeddings that are needed in carrying out the derivations for the proofs of implications explained above. Finally, the statement on associativity of embedding composition $A(n)$ is used in a similar manner. All these statements are defined formally later in this section. For ease of reference we again list these definitions together in Appendix A.

The formal inductive proof that the conjunction of all these statements holds for all integers $n > 0$ is achieved in three steps:

(1) Showing that base cases *i.e.* statements for $n = 0$ or $n = 1$ hold.

(2) Showing that a series of implications is true, *i.e.* for each implication a conjunction of a subset of these statements for $n-1$ implies that the statement for n holds.

(3) Combining all these implications together into an inductive proof.

The same proof technique is later used to show that a *composite* of two diagrams is well-defined in Theorem 2.4.19, as well as to prove some results on associativity and distributivity of diagram composition in Lemmas 2.5.7 and 2.5.13.

Firstly, let us introduce the auxiliary logical statements dependent on the diagram dimension n. The main statement is on well-definedness of a rewritten n-diagram.

Definition 2.3.8 ($R(n)$)**.** For $n \geq 0$, let $R(n)$ denote the statement that for any well-defined n-diagrams D, S, T such that S, T are globular with respect to each other, and a well-defined embedding $e : S \hookrightarrow D$, the rewrite $D.\Pi[e,T]$ of D by e is a well-defined diagram.

The second statement is a result on globularity of individual slices in a well-defined diagram. This is a necessary condition to be able to consider them as rewrites of a well-defined $(n-1)$-diagram.

Definition 2.3.9 ($T(n)$)**.** For $n \geq 2$, let $T(n)$ denote the statement that for any well-defined n-diagram D, we have $(D.s).s = (D[j].d).s$ and $(D.s).t = (D[j].d).t$ for any $0 \leq j < |D|$.

The next statement allows us to express slices of a rewritten diagram in an explicit way instead of depending on the recursive definition. This alternative expression plays the pivotal role in the recursive proof of well-definedness of a rewritten diagram.

Definition 2.3.10 ($S(n)$)**.** For $n \geq 1$, let $S(n)$ denote the statement that for any well-defined n-diagrams D, S, T such that $s(S) = s(T)$, $t(S) = t(T)$ and a well-defined n-diagram embedding $e : S \hookrightarrow A$ the following hold:

$$A.\Pi[e,T][j].d = \\ = \begin{cases} A[j].d & \text{if } 0 \leq j \leq e.h \\ A[e.h].d.\Pi[e.e, T[j-e.h].d] & \text{if } e.h \leq j \leq e.h+|T| \\ A[j+|S|-|T|].d & \text{if } e.h+|T| \leq j < |A|-|S|+|T| \end{cases}$$

The first and the last slice of the middle segment remain unchanged because of the globularity conditions imposed on S, T. For that reason the endpoints of the intervals in the definition can overlap, an additional benefit is that it makes some of the technical proofs simpler.

An interesting question is why not to make the above statement the *definition* of rewriting. The reasons are as follows: Note that our diagrams are given in terms of sources, generators and embeddings, and a slice is a derived concept. Given a list of slices for the diagram, we can deduce the primitive data, but it is not given explicitly. We could in principle say that the rewritten diagram is such, that its data is derived from the given list of slices, but in fact, the data that we would have obtained is exactly the data that we give in Definition 2.3.4. While, the formulation given by $S(n)$ is a useful auxiliary tool, making it the formal definition of rewriting would introduce unnecessary complexity into the theoretical setup.

The next two auxiliary statements let us respectively, decompose a lifted embedding into a composite of two individual lifted embeddings and express a composite rewrite in two equivalent manners. These are both indirectly used to establish the statement $S(n)$ on expressing slices of a rewritten n-diagram as rewrites of a well-defined $(n-1)$-diagram.

Definition 2.3.11 ($Q(n)$)**.** For $n \geq 0$, let $Q(n)$ denote the statement that for any well-defined n-diagrams A, B, C, S, T such that pairs S, T and A, C are globular with respect to each other and for well-defined embeddings $e : S \hookrightarrow A$, $f : C \hookrightarrow B$, the following holds:

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

Definition 2.3.12 ($P(n)$)**.** For $n \geq 0$, let $P(n)$ denote the statement that for any well-defined n-diagrams S, T, A, B, C such that pairs S, T and A, C are globular with respect to each other and for well-defined embeddings $e : S \hookrightarrow A$, $f : C \hookrightarrow B$, the following holds for $0 \leq j \leq e.h$:

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

The auxiliary concept of a lifted embedding is used in constructing the rewritten diagram and as such has to itself be a well-defined embedding. Similarly for the composite of two well-defined embeddings.

Definition 2.3.13 ($B(n)$)**.** For $n \geq 0$, let $B(n)$ denote the statement that for any well-defined n-diagrams S, T, A such that S, T are globular with respect to each other and for a well-defined embedding $e : S \hookrightarrow A$, then the lifted embedding $e.\Lambda[T] : T \hookrightarrow A.\Pi[e, T]$ is well-defined.

Definition 2.3.14 ($C(n)$)**.** For $n \geq 0$, let $C(n)$ denote the statement that given two n-diagram embeddings $e : S \hookrightarrow D$ and $f : D \hookrightarrow M$ between well-defined n-diagrams S, D, M, their composite $f \circ e : S \hookrightarrow M$ is well-defined.

Intuitively, given four diagrams S, D, M, N such that S is embedded in D, D embedded in M and M embedded in N it should not matter in what order we compose these embeddings to obtain an embedding of S in N, hence the final statement is the result on associativity of composition of well-defined embeddings.

Definition 2.3.15 ($A(n)$)**.** For $n \geq 0$, let $A(n)$ denote the statement that given three n-diagram embeddings $e : S \hookrightarrow D$, $f : D \hookrightarrow M$, $g : M \hookrightarrow N$ between well-defined n-diagrams S, D, M, N the following equality holds:

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

To make the following exposition easier to follow, we summarise the main results that are used in the inductive step of the proof of Theorem 2.3.27, which states the conjunction of all the above statements holds for all $n \geq 0$. These implications are as follows:

- $S(n) \wedge T(n) \wedge R(n-1) \implies R(n)$, proved in Lemma 2.3.19.

- $T(n) \wedge P(n-1) \wedge C(n-1) \wedge B(n-1) \implies S(n)$, proved in Lemma 2.3.20.
- $S(n-1) \implies T(n)$, proved in Lemma 2.3.21.
- $S(n) \wedge P(n) \wedge A(n-1) \implies Q(n)$, proved in Lemma 2.3.22.
- $S(n) \wedge Q(n-1) \implies P(n)$, proved in Lemma 2.3.23.
- $R(n) \implies B(n)$, proved in Lemma 2.3.24.
- $C(n-1) \wedge Q(n-1) \wedge A(n-1) \implies C(n)$, proved in Lemma 2.3.25.
- $S(n) \wedge Q(n-1) \wedge A(n-1) \implies A(n)$, proved in Lemma 2.3.26.

We now prove several lemmas establishing the base cases for the main inductive proof.

Lemma 2.3.16. *C(0) holds.*

Proof. We need to show that given two 0-diagram embeddings $e : S \hookrightarrow D$ and $f : D \hookrightarrow M$ between well-defined 0-diagrams S, D, M, their composite $f \circ e : S \hookrightarrow M$ is well-defined.

The domain and codomain of $f \circ e$ are well-defined by assumption. Both f and e consist of no data, so there is nothing to check and the result is vacuously true. □

Lemma 2.3.17. *R(0) holds.*

Proof. We need to show that given well-defined 0-diagrams D, S, T such that S and T are globular with respect to each other, and a well-defined embedding $e : S \hookrightarrow D$, the rewrite $D.\Pi[e, T]$ of D by e is a well-defined diagram.

The 0-diagram embedding e consists of no data. $D.\Pi[e, T]$ is a 0-diagram, its list of generators consists of a single cell and there is no source and no slices, so the result is vacuously true. □

Lemma 2.3.18. *P(0) holds.*

Proof. We need to show that:

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

Since all S, T, A, B, C are all 0-diagrams, their rewrites exist as $R(0)$ holds by Lemma 2.3.17. Since $n = 0$, for the equation to hold, by Definition 2.3.7, we need to check the following:

- Sizes are equal, all steps in this derivation follow by Eq. (2.3) :

$$\begin{aligned} &|B.\Pi[f, A.\Pi[e, T]]| = \\ &\quad = |B| - |C| + |A.\Pi[e, T]| \\ &\quad = |B| - |C| + |A| - |S| + |T| \\ &\quad = |B.\Pi[f, A]| - |S| + |T| \\ &\quad = |(B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]| \end{aligned}$$

  Additionally, since S, T, A, B, C all are well-defined 0-diagrams, then each consists of just a single generator and $|A| = |B| = |C| = |S| = |T| = 1$.

  Then it follows that $|B| - |C| + |A| - |S| + |T| = 1$, and:

$$\begin{aligned} &|(B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]| = \\ &\quad = |B.\Pi[f, A.\Pi[e, T]]| = 1 \end{aligned}$$

  Hence, we only need to check that the single element in the list of generators of each of these two diagrams is the same.

- Generators are equal, all steps in this derivation follow by Eq. (2.4) :

$$\begin{aligned}
&((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T])[0].g = \\
&\quad = T[0].g \\
&\quad = (A.\Pi[e, T])[0].g \\
&\quad = (B.\Pi[f, A.\Pi[e, T]])[0].g
\end{aligned}$$

By this argument the equality holds, hence so does $P(0)$. □

With these base cases established, we now proceed to prove a series of implications between the logical statements defined earlier in this chapter. Each time we only take the minimal subset of expressions for $(n-1)$ that implies the given statement for n.

Lemma 2.3.19. *For $n \geq 1$, the following holds: $S(n) \wedge T(n) \wedge R(n-1) \implies R(n)$. Additionally, if $n = 1$, then $S(n) \wedge R(n-1) \implies R(n)$*

Proof. Let us assume that $S(n)$ and $R(n-1)$ hold, then we need to show $D.\Pi[e, T]$ is well-defined. By Definition 2.2.6, this means that we require:

- $(D.\Pi[e, T]).s$ is well-defined
- $(D.\Pi[e, T])[j].d$ for $0 < j < |D.\Pi[e, T]|$ exists and is well-defined

The first statement follows since:

$$(D.\Pi[e, T]).s = D.s$$

$D.s$ is well-defined as the source of a well-defined diagram D. We prove the second statement separately for three ranges within $0 \leq j \leq |D.\Pi[e, T]|$

- For $0 < j \leq e.h$, by $S(n)$ we obtain that:

  $$D.\Pi[e, T][j].d = D[j].d$$

  Hence all slices in this section exist and are well-defined as D is well-defined.

- For $e.h \leq j < e.h + |T|$, by $S(n)$ we obtain that:

  $$D.\Pi[e, T][j].d = (D[e.h].d).\Pi[e.e, T[j - e.h].d]$$

  ⋆ $D[e.h].d$ is well-defined as the slice of a well-defined diagram D

  ⋆ The embedding $e.e$ is well-defined as e is well-defined

  ⋆ If $n = 1$, $s(S) = S(T)$ and $T[j - e.h].d$ are 0-diagrams and globular with respect to each other

  ⋆ If $n > 1$, we need $s(s(S)) = s(T[j - e.h].d)$ and $t(s(S)) = t(T[j - e.h].d)$. We note that $s(S) = S(T) = T[0].d$, then the result follows by $T(n)$.

  By applying $R(n-1)$ to the rewrite of the $(n-1)$-diagram $D[e.h].d$, we get that the diagram $(D[e.h].d).\Pi[e.e, T[j - e.h].d]$ exists and is well-defined, hence so is $D.\Pi[e, T][j].d$, as required.

- For $e.h + |T| \leq j < |D| - |S| + |T|$, by $S(n)$ we obtain that:

  $$D.\Pi[e, T][j].d = D[j - |T| + |S|].d$$

  It follows that all slices in this section exist and are well-defined, since all $D[j].d$ are well-defined as slices of the well-defined diagram D.

Hence, all the slices $(D.\Pi[e,T])[j].d$ exists and are also well-defined, hence this diagram is itself well-defined. By this $R(n)$ holds, hence the implication is true. Note that $T(n)$ is only used for $n > 1$, hence $S(1) \wedge R(0) \implies R(1)$ holds. ☐

Lemma 2.3.20. *For $n > 1$ the following holds: $T(n) \wedge P(n-1) \wedge C(n-1) \wedge B(n-1) \implies S(n)$. Additionally, for $n = 1$: $P(0) \wedge C(0) \wedge B(0) \implies S(1)$*

Proof. Let us assume that all $T(n)$, $P(n-1)$, $C(n-1)$, $B(n-1)$ hold. We show that $S(n)$ holds in each of the individual three ranges separately:

- For $0 \leq j \leq e.h$:

  In this range, we show this result by induction on j.

  ⋆ *Base case:* For $j = 0$

  $$A.\Pi[e,T][0].d = A.\Pi[e,T].s = A.s = A[0].s$$

  ⋆ *Inductive step:* For $0 < j \leq e.h$ assume by induction that:

  $$A.\Pi[e,T][j].d = A[j].d \qquad (IH)$$

  Let us consider $A.\Pi[e,T][j+1].d$: Note that by Definition 2.3.4 for $0 < j \leq e.h$ we have the following:

  ∗ $A.\Pi[e,T][j].g = A[j].g$
  ∗ $A.\Pi[e,T][j].e = A[j].e$

  Hence:

  $$\begin{aligned} & A.\Pi[e,T][j+1].d \\ &\quad = (A.\Pi[e,T][j].d).\Pi[A.\Pi[e,T][j].e, \\ &\qquad\quad t(A.\Pi[e,T][j].g)] && \text{[Def. 2.2.3]} \\ &\quad = (A.\Pi[e,T][j].d).\Pi[A[j].e, t(A[j].g)] && \text{[Def. 2.3.4]} \\ &\quad = (A[j].d).\Pi[A[j].e, t(A[j].g)] && [IH] \\ &\quad = A[j+1].d && \text{[Def. 2.2.3]} \end{aligned}$$

  As required.

  By induction we have that: $A.\Pi[e,T][j].d = A[j].d$ holds for $0 \leq j \leq e.h$.

- For $e.h \leq j \leq e.h + |T|$:

  First, for the rewrite $(A[e.h].d).\Pi[e.e, T[j-e.h].d]$ to be well-defined, we need to show that the source $s(S)$ and the target $T[j-e.h].d$ of the rewrite are globular with respect to each other.

  ⋆ For $n = 1$, both $s(S)$ and $T[j-e.h].d$ are 0-diagrams for $e.h \leq j \leq |e.h + |T|$ so there are no conditions to check.

  ⋆ For $n > 1$, we need $s(s(S)) = s(T[j-e.h].d)$ and $t(s(S)) = t(T[j-e.h].d)$ for $e.h \leq j \leq |e.h + |T|$.

  Note that since S and T are globular, we have:

  $$s(S) = s(T) = T[0].d = T.s$$

  We then apply proposition $T(n)$ to the diagram T, to obtain the necessary result.

  We can now prove this result by induction on $e.h \leq j \leq e.h + |T|$:

- ⋆ *Base case:* For $j = e.h$

$$\begin{aligned}
&(A[e.h].d).\Pi[e.e, T[j - e.h].d] \\
&\quad = (A[e.h].d).\Pi[e.e, T[e.h - e.h].d] && [j = e.h] \\
&\quad = (A[e.h].d).\Pi[e.e, T[0].d] \\
&\quad = (A[e.h].d).\Pi[e.e, s(T)] && [\text{Def. 2.2.5}] \\
&\quad = (A[e.h].d).\Pi[e.e, s(S)] && [t(S) = t(T)] \\
&\quad = A[e.h].d && [\text{Lemma 2.3.5}] \\
&\quad = A.\Pi[e, T][e.h].d
\end{aligned}$$

  The final transformation follows by this result for the initial range $0 \leq j \leq e.h$.

- ⋆ *Inductive step:* For $e.h < j \leq e.h + |T|$ assume by induction that:

$$A.\Pi[e, T][j].d = A[e.h].d.\Pi[e.e, T[j - e.h].d] \qquad (IH)$$

  Let us consider $A.\Pi[e, T][j + 1].d$:

$$\begin{aligned}
&A.\Pi[e, T][j + 1].d \\
&\quad = (A.\Pi[e, T][j].d).\Pi[A.\Pi[e, T][j].e, \\
&\qquad\quad t(A.\Pi[e, T][j].g)] && [\text{Def. 2.2.3}] \\
&\quad = (A.\Pi[e, T][j].d).\Pi[(e.e).\Lambda[T[j - e.h].d] \\
&\qquad\quad \circ T[j - e.h].e, t(T[j - e.h].g)] && [\text{Def. 2.3.4}] \\
&\quad = ((A[e.h].d).\Pi[e.e, T[j - e.h].d]).\Pi[(e.e).\Lambda[T[j - e.h].d] \\
&\qquad\quad \circ T[j - e.h].e, t(T[j - e.h].g)] && [IH] \\
&\quad = (A[e.h].d).\Pi[e.e, (T[j - e.h].d).\Pi[T[j - e.h].e, \\
&\qquad\quad t(T[j - e.h].g)]] && [P(n - 1)] \\
&\quad = (A[e.h].d).\Pi[e.e, T[(j + 1) - e.h].d] && [\text{Def. 2.2.3}]
\end{aligned}$$

  $P(n - 1)$ may be applied here as all the embeddings involved in the rewrites are well-defined. The lifted embedding between $(n - 1)$-diagrams is well-defined as $B(n - 1)$ holds and the composed embedding of $(n - 1)$-diagram embeddings is well-defined, as $C(n - 1)$ holds.

- For $e.h + |T| \leq j < |A| - |S| + |T|$:

  We show the result in this range by induction on j.

  - ⋆ *Base case:* For $j = e.h + |T|$, we have:

$$\begin{aligned}
&A.\Pi[e, T][e.h + |T|].d \\
&\quad = (A[e.h].d).\Pi[e.e, T[(e.h + |T|) - e.h]] \\
&\quad = (A[e.h].d).\Pi[e.e, T[|T|]] \\
&\quad = (A[e.h].d).\Pi[e.e, t(T)]] && [\text{Def. 2.2.5}] \\
&\quad = (A[e.h].d).\Pi[e.e, t(S)]] && [t(S) = t(T)] \\
&\quad = (A[e.h].d).\Pi[e.e, S[|S|]] && [\text{Def. 2.2.5}] \\
&\quad = A.\Pi[e, S][e.h + |S|].d && [\text{Def. 2.2.3}] \\
&\quad = A[e.h + |S|].d && [\text{Lemma 2.3.5}]
\end{aligned}$$

    The penultimate transformation follows by the result for the range $e.h \leq j \leq e.h + |S|$ applied to the identity rewrite of A $e : S \hookrightarrow A$.

* ⋆ *Inductive step:* For $e.h + |T| < j \leq |A| + |T| - |S|$, note that, similarly as in the range $0 \leq j \leq e.h$, we have that by Definition 2.3.4:
  * ∗ $A.\Pi[e, T][j].g = A[j + |S| - |T|.g$
  * ∗ $A.\Pi[e, T][j].e = A[j + |S| - |T|.e$

  The rest of the argument is the same as for the range $0 \leq j \leq e.h$.

  By induction we have proved that: $A.\Pi[e, T][j].d = A[j].d$ holds for $e.h + |T| \leq j \leq |A| + |T| - |S|$.

By this, $S(n)$ holds for each of the three ranges, hence the implication is true. Note that the assumption $T(n)$ is only used to show globularity for $n > 1$, hence we also have $P(0) \wedge C(0) \wedge B(0) \implies S(1)$.

□

Lemma 2.3.21. *For $n \leq 2$ the following holds: $S(n-1) \implies T(n)$*

Proof. Let us assume that $S(n-1)$ holds. We need to show that for any well-defined n-diagram D, we have $(D.s).s = (D[j].d).s$ and $(D.s).t = (D[j].d).t$ for any $0 \leq j < |D|$.

First, we prove this result by induction on i:

* *Base case*: For $j = 0$, by Definition 2.3.4, we have: $D.s = D[0].d$, so the result trivially holds.
* *Inductive step*: For $j > 0$, let us assume by induction that:

$$(D.s).s = (D[j].d).s \qquad (IH(\text{a}))$$
$$(D.s).t = (D[j].d).t \qquad (IH(\text{b}))$$

Firstly, let us consider $(D[i+1].d).s$:

$$\begin{aligned}
&(D[j+1].d).s = \\
&\quad = ((D[j].d).\Pi[D[j].e, t(D[j].g)]).s && \text{[Def. 2.2.3]} \\
&\quad = (D[j].d).s && \text{[Def. 2.3.4]} \\
&\quad = (D.s).s && [IH(\text{a})]
\end{aligned}$$

Secondly, let us consider $(D[i+1].d).t$:

$$\begin{aligned}
&(D[i+1].d).t = \\
&\quad = ((D[j].d).\Pi[D[j].e), t(D[j].g)]).t && \text{[Def. 2.2.3]} \\
&\quad = ((D[j].d).\Pi[D[j].e), \\
&\qquad\quad t(D[j].g)])[|D[j].d| - |s(D[j].g)| + |t(D[j].g)|].d && \text{[Def. 2.2.5]} \\
&\quad = (D[j].d)[(|D[j].d| - |s(D[j].g)| + |t(D[j].g)|) \\
&\qquad\quad + |s(D[j].g)| - |s(D[j].g)|] && [S(n-1)] \\
&\quad = (D[j].d)[|D[j].d|] \\
&\quad = (D[j].d).t && \text{[Def. 2.2.5]} \\
&\quad = (D.s).t && [IH(b)]
\end{aligned}$$

By this, the result holds for all $0 \leq i \leq |D|$, hence $T(n)$ holds and the implication is true. □

Lemma 2.3.22. *For $n > 0$ the following holds: $S(n) \wedge P(n) \wedge A(n-1) \implies Q(n)$. Additionally, for $n = 0$: $P(0) \implies Q(0)$*

Proof. Let us assume that $S(n)$, $P(n)$ and $A(n-1)$ all hold. By Lemma 2.3.19 the statement $R(n)$ also holds and all the following n-diagram rewrites are well-defined. We need to show that the following two n-diagram embeddings are equivalent:

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

For this, by Definition 2.3.6 domains and codomains must be equivalent diagrams. By Definition 2.3.1, types of the individual embeddings are as follows:

$$\begin{aligned}
&f.\Lambda[A] \circ e : S \hookrightarrow B.\Pi[f, A] \\
&(f.\Lambda[A] \circ e).\Lambda[T] : T \hookrightarrow B.\Pi[f, A].\Pi[f.\Lambda[A] \circ e, T] \\
&f.\Lambda[A.\Pi[e, T]] : A.\Pi[e, T] \hookrightarrow B.\Pi[f, A.\Pi[e, T]] \\
&e.\Lambda[T] : T \hookrightarrow A.\Pi[e, T]
\end{aligned}$$

As the domain of $f.\Lambda[A.\Pi[e, T]]$ matches the codomain of $e.\Lambda[T]$, we can conclude that the composite $f.\Lambda[A.\Pi[e, T]] \circ e.\Lambda[T]$ exists. The domain of this composite matches the domain of $(f.\Lambda[A] \circ e).\Lambda[T]$.

The codomain of $f.\Lambda[A.\Pi[e, T]] \circ e.\Lambda[T]$ is $B.\Pi[f, A.\Pi[e, T]]$, the codomain of $(f.\Lambda[A] \circ e).\Lambda[T]$ is $B.\Pi[f, A].\Pi[f.\Lambda[A] \circ e, T]$. These two diagrams are equivalent by $P(n)$, hence the codomains of both embeddings also match, as required.

If $n > 0$, then we need to check two additional conditions:

- Component embeddings are equivalent:

$$\begin{aligned}
&(f.\Lambda[A.\Pi[e, T]] \circ e.\Lambda[T]).e \\
&\quad = (f.\Lambda[A.\Pi[e, T]].e).\Lambda[A.\Pi[e, T][e.\Lambda[T].h].d] \circ \\
&\qquad\quad e.\Lambda[T].e && \text{[Eq. (2.8)]} \\
&\quad = (f.\Lambda[A.\Pi[e, T]].e).\Lambda[A.\Pi[e, T][e.\Lambda[T].h].d] \circ e.e && \text{[Def. 2.3.1]} \\
&\quad = (f.\Lambda[A.\Pi[e, T]].e).\Lambda[A.\Pi[e, T][e.h].d] \circ e.e && \text{[Def. 2.3.1]} \\
&\quad = (f.\Lambda[A.\Pi[e, T]].e).\Lambda[A[e.h].d] \circ e.e && [S(n)] \\
&\quad = (f.e).\Lambda[A[e.h].d] \circ e.e && \text{[Def. 2.3.1]} \\
&\quad = (f.\Lambda[A].e).\Lambda[A[e.h].d] \circ e.e && \text{[Def. 2.3.1]} \\
&\quad = (f.\Lambda[A] \circ e).e && \text{[Def. 2.3.2]} \\
&\quad = ((f.\Lambda[A] \circ e).\Lambda[T]).e && \text{[Def. 2.3.1]}
\end{aligned}$$

- Heights are equal:

$$\begin{aligned}
&(f.\Lambda[A.\Pi[e, T]] \circ e.\Lambda[T]).h \\
&(f.\Lambda[A.\Pi[e, T]]).h + (e.\Lambda[T]).h && \text{[Def. 2.3.2]} \\
&\quad = f.h + e.h && \text{[Def. 2.3.1]} \\
&\quad = (f.\Lambda[A]).h + e.h && \text{[Def. 2.3.1]} \\
&\quad = (f.\Lambda[A] \circ e).h && \text{[Def. 2.3.2]} \\
&\quad = ((f.\Lambda[A] \circ e).\Lambda[T]).h && \text{[Def. 2.3.1]}
\end{aligned}$$

Both conditions are satisfied, hence these two embeddings are equivalent and $Q(n)$ holds. By this, the implication $S(n) \wedge P(n) \implies Q(n)$ is true for $n > 0$. Note that the assumptions $S(n)$, $A(n-1)$ are only used when $n > 0$, hence $P(0) \implies Q(0)$ also holds.

□

Lemma 2.3.23. *For $n \geq 1$ the following holds: $S(n) \wedge Q(n-1) \implies P(n)$.*

Proof. Let us assume that both $S(n)$ and $Q(n-1)$ hold. By Lemma 2.3.19 the statement $R(n)$ also holds and all the following n-diagram rewrites are well-defined.

We need to show that for well-defined diagrams S, T, A, B, C such that $s(S) = s(T)$, $t(S) = t(T)$, $s(A) = s(C)$, $t(A) = t(C)$ and well-defined embeddings $e : S \hookrightarrow A$, $f : C \hookrightarrow B$, the following holds:

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

By Definition 2.3.7, we need to check the following four conditions:

- Sources are equivalent diagrams: All equalities in this derivation follow by the source clause Eq. (2.5) in the definition of a rewrite.

$$\begin{aligned}
&((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).s = \\
&\quad = (B.\Pi[f, A]).s \\
&\quad = B.s \\
&\quad = (B.\Pi[f, A.\Pi[e, T]]).s
\end{aligned}$$

- Sizes of lists of generators are equal: All equalities in this derivation follow by the size clause Eq. (2.6) in the definition of a rewrite.

$$\begin{aligned}
&|B.\Pi[f, A.\Pi[e, T]]| = \\
&\quad = |B| - |C| + |A.\Pi[e, T]| \\
&\quad = |B| - |C| + |A| - |S| + |T| \\
&\quad = |B.\Pi[f, A]| - |S| + |T| \\
&\quad = |(B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]|
\end{aligned}$$

- Generators are equal, for $0 \leq i \leq |B.\Pi[f, A.\Pi[e, T]]|$. All derivations in this section follow by the generator clause Eq. (2.7) in the definition of a rewrite.

  We consider i in five separate ranges:

  - ⋆ $0 \leq i \leq f.h$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e, T]])[i].g = \\
&\quad = B[i].g \\
&\quad = (B.\Pi[f, A])[i].g \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).g
\end{aligned}$$

  - ⋆ $f.h \leq i \leq f.h + e.h$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e, T]])[i].g = \\
&\quad = (A.\Pi[e, T])[i - f.h].g \\
&\quad = A[i - f.h].g \\
&\quad = (B.\Pi[f, A])[i].g \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).g
\end{aligned}$$

⋆ $f.h + e.h \leq i \leq f.h + e.h + |T|$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e, T]])[i].g = \\
&\quad = (A.\Pi[e, T])[i - f.h].g \\
&\quad = T[(i - f.h) - e.h].g \\
&\quad = T[(i - (f.h + e.h)].g \\
&\quad = T[i - (f.\Lambda[A].h + e.h)].g \\
&\quad = T[i - (f.\Lambda[A] \circ e).h].g \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).g
\end{aligned}$$

⋆ $f.h + e.h + |T| \leq i \leq f.h + |A| - |S| + |T|$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e, T]])[i].g = \\
&\quad = A.\Pi[e, T])[i - f.h].g \\
&\quad = A[(i - f.h) - |T| + |S|].g \\
&\quad = A[(i - |T| + |S|) - f.h].g \\
&\quad = B.\Pi[f, A][i - |T| + |S|].g \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).g
\end{aligned}$$

⋆ $f.h + |A| - |S| + |T| \leq i \leq |B| - |C| + |A| - |S| + |T|$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e, T]])[i].g \\
&\quad = B[i - |A.\Pi[e, T]| + |C|].g \\
&\quad = B[i - |A| + |C| - |T| + |S|].g \\
&\quad = (B.\Pi[f, A])[i - |T| + |S|].g \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).g
\end{aligned}$$

- Embeddings are equivalent, similarly as for generators, for $0 \leq i \leq |B.\Pi[f, A.\Pi[e, T]]|$. We consider i in five separate ranges:

⋆ $0 \leq i \leq f.h$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e, T]])[i].e \\
&\quad = B[i].e && \text{[Eq. (2.8)]} \\
&\quad = (B.\Pi[f, A])[i].e && \text{[Eq. (2.8)]} \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).e && \text{[Eq. (2.8)]}
\end{aligned}$$

⋆ $f.h \leq i \leq f.h + e.h$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e, T]])[i].e \\
&\quad = (f.e).\Lambda[A.\Pi[e, T][i - f.h].d] \circ A.\Pi[e, T][i - f.h].e && \text{[Eq. (2.8)]} \\
&\quad = (f.e).\Lambda[A.\Pi[e, T][i - f.h].d] \circ A[i - f.h].e && \text{[Eq. (2.8)]} \\
&\quad = (f.e).\Lambda[A[i - f.h].d] \circ A[i - f.h].e && [S(n)] \\
&\quad = (B.\Pi[f, A])[i].e && \text{[Eq. (2.8)]} \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).e && \text{[Eq. (2.8)]}
\end{aligned}$$

$\star$ $f.h + e.h \leq i \leq f.h + e.h + |T|$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e,T]])[i].e \\
&\quad = (f.e).\Lambda[A.\Pi[e,T][i - f.h].d] \circ A.\Pi[e,T][i - f.h].e && \text{[Eq. (2.8)]} \\
&\quad = (f.e).\Lambda[A.\Pi[e,T][i - f.h].d] \\
&\qquad\quad \circ ((e.e).\Lambda[T[(i - f.h) - e.h].d] \circ T[(i - f.h) - e.h].e) && \text{[Eq. (2.8)]} \\
&\quad = ((f.e).\Lambda[A.\Pi[e,T][i - f.h].d] \\
&\qquad\quad \circ (e.e).\Lambda[T[(i - f.h) - e.h].d]) \circ T[(i - f.h) - e.h].e && [A(n-1)] \\
&\quad = ((f.e).\Lambda[A[e.h].\Pi[e.e, T[i - f.h]]] \\
&\qquad\quad \circ (e.e).\Lambda[T[(i - f.h) - e.h].d]) \circ T[(i - f.h) - e.h].e && [S(n)] \\
&\quad = (((f.e).\Lambda[A[e.h].d] \circ e.e).\Lambda[T[(i - f.h) - e.h]] \\
&\qquad\quad \circ T[(i - f.h) - e.h].e && [Q(n-1)] \\
&\quad = ((f.e).\Lambda[A[e.h].d] \circ e.e).\Lambda[T[i - (f.h + e.h)]] \\
&\qquad\quad \circ T[i - (f.h + e.h)].e \\
&\quad = ((f.\Lambda[A].e).\Lambda[A[e.h].d] \circ e.e).\Lambda[T[i - (f.h + e.h)]] \\
&\qquad\quad \circ T[i - (f.h + e.h)].e && \text{[Def. 2.3.1]} \\
&\quad = ((f.\Lambda[A] \circ e).e).\Lambda[T[i - (f.h + e.h)]] \\
&\qquad\quad \circ T[i - (f.h + e.h)].e && \text{[Def. 2.3.2]} \\
&\quad = ((f.\Lambda[A] \circ e).e).\Lambda[T[i - (f.\Lambda[A].h + e.h)]] \\
&\qquad\quad \circ T[i - (f.\Lambda[A].h + e.h)].e && \text{[Def. 2.3.1]} \\
&\quad = ((f.\Lambda[A] \circ e).e).\Lambda[T[i - (f.\Lambda[A] \circ e).h]] \\
&\qquad\quad \circ T[i - (f.\Lambda[A] \circ e).h].e && \text{[Def. 2.3.2]} \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).g && \text{[Eq. (2.8)]}
\end{aligned}$$

$\star$ $f.h + e.h + |T| \leq i \leq f.h + |A| - |S| + |T|$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e,T]])[i].g \\
&\quad = (f.e).\Lambda[A.\Pi[e,T][i - f.h].d] \\
&\qquad\quad \circ A.\Pi[e,T][i - f.h].e && \text{[Eq. (2.8)]} \\
&\quad = (f.e).\Lambda[A.\Pi[e,T][i - f.h].d] \\
&\qquad\quad \circ A[(i - |T| + |S|) - f.h].e && \text{[Eq. (2.8)]} \\
&\quad = (f.e).\Lambda[A[(i - |T| + |S|) - f.h].d] \\
&\qquad\quad \circ A[(i - |T| + |S|) - f.h].e && [S(n)] \\
&\quad = B.\Pi[f, A][i - |T| + |S|].e && \text{[Eq. (2.8)]} \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).e && \text{[Eq. (2.8)]}
\end{aligned}$$

$\star$ $f.h + |A| - |S| + |T| \leq i \leq |B| - |C| + |A| - |S| + |T|$

$$\begin{aligned}
&(B.\Pi[f, A.\Pi[e,T]])[i].e \\
&\quad = B[i - |A.\Pi[e,T]| + |C|].e && \text{[Eq. (2.8)]} \\
&\quad = B[i - |A| + |C| - |T| + |S|].e && \text{[Def. 2.3.4]} \\
&\quad = (B.\Pi[f, A])[i - |T| + |S|].e && \text{[Eq. (2.8)]} \\
&\quad = ((B.\Pi[f, A]).\Pi[f.\Lambda[A] \circ e, T]).e && \text{[Eq. (2.8)]}
\end{aligned}$$

$\square$

Lemma 2.3.24. *For $n \geq 1$ the following holds: $R(n) \implies B(n)$.*

Proof. Let us assume that $R(n)$ holds. We need to show that $e.\Lambda[T] : T \hookrightarrow A.\Pi[e, T]$ is well-defined. For that, the domain and codomain of $e.\Lambda[T]$ have to be well-defined. T is well-defined by assumption, $A.\Pi[e, T]$ is well-defined by $R(n)$.

- If $n = 0$, we need $A.\Pi[e, T][0].g = T[0].g$, but this follows directly from the definition of a rewrite for $n = 0$.
- If $n > 0$, by Definition 2.2.8 we need to show three separate conditions:
  - ⋆ $e.\Lambda[T].e$ is well-defined.
    By Definition 2.3.1, we have $e.\Lambda[T].e = e.e$. As e is well-defined, so is its component embedding $e.e$. Hence, $e.\Lambda[T].e$ is equivalent to a well-defined embedding and itself is well-defined.
  - ⋆ For every $0 \leq i < |T|$ we need to show that the generators of both diagrams satisfy the following:

$$\begin{aligned} & A.\Pi[e, T][i + e.\Lambda[T].h].g \\ & \quad = A.\Pi[e, T][i + e.h].g && \text{[Def. 2.3.1]} \\ & \quad = T[(i + e.h) - e.h].g && \text{[Eq. (2.8)]} \\ & \quad = T[i].g \end{aligned}$$

  - ⋆ For every $0 \leq i < |T|$ we need to show that the embeddings of both diagrams satisfy the following:

$$\begin{aligned} & A.\Pi[e, T][i + e.\Lambda[T].h].e \\ & \quad = A.\Pi[e, T][i + e.h].e && \text{[Def. 2.3.1]} \\ & \quad = (e.e).\Lambda[T[i].d] \circ T[i].e && \text{[Def. 2.8]} \\ & \quad = (e.\Lambda[T].e).\Lambda[T[i].d] \circ T[i].e && \text{[Def. 2.3.1]} \end{aligned}$$

As all three conditions are satisfied, $e.\Lambda[T]$ is a well-defined embedding and $B(n)$ holds, hence the implication is true. □

Lemma 2.3.25. *For $n \geq 1$ the following holds: $C(n-1) \wedge Q(n-1) \wedge A(n-1) \implies C(n)$.*

Proof. We assume that all $C(n-1)$, $A(n-1)$ and $S(n-1)$ hold. By combining Lemmas 2.3.24 and 2.3.19, we have that since $S(n-1)$ holds, both $R(n-1)$ and $B(n-1)$ also hold.

We need to show that given two n-diagram embeddings $e : S \hookrightarrow D$ and $f : D \hookrightarrow M$ between well-defined n-diagrams S, D, M, their composite $f \circ e : S \hookrightarrow M$ is well-defined.

The domain and codomain of $f \circ e$ are well-defined by assumption. To show that $f \circ e$ is well-defined, by Definition 2.2.8 we need to show three separate conditions:

- $(f \circ e).e$ is well-defined.

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

  Both $e.e$ and $f.e$ are well-defined $(n-1)$-embeddings, as they are component embeddings of well-defined embeddings f, e. $(f.e).\Lambda[D[e.h].d]$ is well-defined $(n-1)$-embedding by $B(n-1)$.

  We then apply the inductive hypothesis to conclude that $(f \circ e).e$ is well-defined.

- For $0 \leq i < |S|$ we have:

$$\begin{aligned} & S[i].g \\ & \quad = D[e.h + i].g && \text{[Eq. (2.1) for } S, D\text{]} \\ & \quad = M[e.h + f.h + i].g && \text{[Eq. (2.1) for } D, M\text{]} \end{aligned}$$

- For $0 \leq i < |S|$ we have:

$$
\begin{aligned}
&M[(f \circ e).h + i].e \\
&\quad = M[(f.h + e.h) + i].e && \text{[Def. 2.3.2]} \\
&\quad = (f.e).\Lambda[D[e.h + i].d] \circ D[e.h + i].e && \text{[Eq. (2.2) for } D, M\text{]} \\
&\quad = (f.e).\Lambda[D[e.h + i]] \circ ((e.e).\Lambda[S[i].d] \circ S[i].e) && \text{[Eq. (2.2) for } S, D\text{]} \\
&\quad = ((f.e).\Lambda[D[e.h + i]] \circ (e.e).\Lambda[S[i].d] \circ) S[i].e && [A(n-1)] \\
&\quad = ((f.e).\Lambda[D[e.h]] \circ e.e).\Lambda[S[i].d] \circ S[i].e && [Q(n-1)] \\
&\quad = ((f \circ e).e).\Lambda[S[i].d] \circ S[i].e && \text{[Def. 2.3.2]}
\end{aligned}
$$

As all conditions of Definition 2.3.7 are satisfied, the two diagrams are equivalent and the result holds.

□

Lemma 2.3.26. *For $n \geq 1$ the following holds: $S(n) \wedge Q(n-1) \wedge A(n-1) \implies A(n)$. Additionally, if $n = 0$, then $A(0)$ holds with no further assumptions.*

Proof. We assume that all $S(n)$, $Q(n-1)$ and $A(n-1)$ hold.

We need to show that the following two n-diagram embeddings are equivalent:

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

For that by Definition 2.3.6 domains and codomains must be equivalent diagrams. By Definition 2.3.2, types of the individual embeddings are as follows:

$$
\begin{aligned}
f \circ e &: S \hookrightarrow M \\
g \circ f &: D \hookrightarrow N \\
g \circ (f \circ e) &: S \hookrightarrow N \\
(g \circ f) \circ e &: S \hookrightarrow N
\end{aligned}
$$

By this we can conclude that the domains and codomains of both embeddings match.

If $n > 0$, then we need to check two additional conditions:

- Heights are equal:

$$
\begin{aligned}
&(g \circ (f \circ e)).h \\
&\quad = g.h + (f \circ e).h && \text{[Def. 2.3.2]} \\
&\quad = g.h + f.h + e.h && \text{[Def. 2.3.2]} \\
&\quad (g \circ f).h + e.h && \text{[Def. 2.3.2]} \\
&\quad ((g \circ f) \circ e).h && \text{[Def. 2.3.2]}
\end{aligned}
$$

- Component embeddings are equivalent:

$$
\begin{aligned}
&(g \circ (f \circ e)).e \\
&\quad = (g.e).\Lambda[M[(f \circ e).h].d] \circ (f \circ e).e && \text{[Def. 2.3.2]} \\
&\quad = (g.e).\Lambda[M[(f \circ e).h].d] \circ ((f.e).\Lambda[D[e.h].d] \circ e.e) && \text{[Def. 2.3.2]} \\
&\quad = ((g.e).\Lambda[M[(f \circ e).h].d] \circ (f.e).\Lambda[D[e.h].d]) \circ e.) && [A(n-1)] \\
&\quad = ((g.e).\Lambda[M[f.h + e.h].d] \circ (f.e).\Lambda[D[e.h].d]) \circ e.) && \text{[Def. 2.3.2]} \\
&\quad = ((g.e).\Lambda[M.\Pi[f, D][f.h + e.h].d] \circ (f.e).\Lambda[D[e.h].d]) \circ e.) && \text{[Def. 2.3.5]} \\
&\quad = ((g.e).\Lambda[M[f.h].d.\Pi[f.e, D[e.h].d]] \circ (f.e).\Lambda[D[e.h].d]) \circ e.) && [S(n)] \\
&\quad = (g.e.\Lambda[M[f.h].d] \circ f.e).\Lambda[D[e.h].d] \circ e.e && [Q(n-1)] \\
&\quad = ((g \circ f).e).\Lambda[D[e.h].d] \circ e.e && \text{[Def. 2.3.2]} \\
&\quad = ((g \circ f) \circ e).e && \text{[Def. 2.3.2]}
\end{aligned}
$$

As all these conditions hold, both embeddings are equivalent, hence $A(n)$ holds and the implication is true. In addition, we only used the assumptions for $n > 0$, hence A(0) also holds with no further requirements. □

With all these implications proved we can now bring them together to prove that the conjunction of selected logical statements holds for all $n \geq 0$.

Theorem 2.3.27. *The statement $S(n) \wedge R(n) \wedge P(n) \wedge C(n) \wedge A(n) \wedge Q(n)$ holds for all $n \geq 1$.*

Proof. We prove the result by induction on n.

- *Base case*: For $n = 1$:
  - ⋆ To establish $S(1)$, by Lemma 2.3.20, we need $P(0)$ to hold. $P(0)$ holds with no further conditions by Lemma 2.3.18.
  - ⋆ To establish $R(1)$, by Lemma 2.3.19, we need $S(1)$ and R(0) to hold. $S(1)$ is given by the statement above. $R(0)$ holds with no further conditions by Lemma 2.3.17.
  - ⋆ To establish $P(1)$, by Lemma 2.3.23, we need $Q(0)$ and $S(1)$ to hold. $Q(0)$ holds, by Lemma 2.3.22 since $P(0)$ holds.
  - ⋆ To establish $C(1)$, by Lemma 2.3.25, we need $C(0)$, $Q(0)$ and $A(0)$ to hold. $C(0)$ holds with no further conditions by Lemma 2.3.16, similarly $A(0)$ by Lemma 2.3.25. $Q(0)$ holds by the statement above.
  - ⋆ To establish $A(1)$, by Lemma 2.3.26, we need $S(1)$ and $Q(0)$ and $A(0)$ to hold. The first two hold by the statements above, $A(0)$ holds by Lemma 2.3.26.
  - ⋆ To establish $Q(1)$, by Lemma 2.3.22, we need $S(1)$ and $P(1)$ and $A(0)$ to hold. The first two hold by the statements above, $A(0)$ holds by Lemma 2.3.26.
- *Inductive step*: For $n > 1$, we assume that all $S(n-1)$, $R(n-1)$, $P(n-1)$, $C(n-1)$, $A(n-1)$, $Q(n-1)$ hold.
  - ⋆ To establish $S(n)$, by Lemma 2.3.20, we need $P(n-1)$, $C(n-1)$, $B(n-1)$ and $T(n)$ to hold. Since, $S(n-1)$ holds, by Lemma 2.3.21 we obtain that $T(n)$ holds. Since $R(n-1)$ holds by Lemma 2.3.24 we obtain that $B(n-1)$ holds. $P(n-1)$ and $C(n-1)$ hold by the inductive hypothesis.
  - ⋆ To establish $R(n)$, by Lemma 2.3.19, we need $S(n)$, $T(n)$ and $R(n-1)$. We have that $S(n)$ and $T(n)$ hold by the statement above. $R(n-1)$ holds by the inductive hypothesis.
  - ⋆ To establish $P(n)$, by Lemma 2.3.23, we need $S(n)$ and $Q(n-1)$. $S(n)$ holds by the earlier part of this argument. $Q(n-1)$ holds by the inductive hypothesis.
  - ⋆ To establish $Q(n-1)$, by Lemma 2.3.22, we need $S(n-1)$ and $P(n-1)$ to hold, both hold by the inductive hypothesis.
  - ⋆ To establish $C(n)$, by Lemma 2.3.25, we need $C(n-1)$, $A(n-1)$ and $Q(n-1)$ to hold. All hold by the inductive hypothesis.
  - ⋆ To establish $A(n)$, by Lemma 2.3.26, we need $S(n)$, $Q(n-1)$ and $A(n-1)$ to hold. $S(n)$ holds by the statements above. $Q(n-1)$ and $A(n-1)$ hold by the inductive hypothesis.
  - ⋆ To establish $Q(n)$, by Lemma 2.3.22, we need $S(n)$, $P(n)$ and $A(n-1)$ to hold. $S(n)$ and $P(n)$ hold by the statements above. $A(n-1)$ holds by the inductive hypothesis.

  Hence, we obtain that $S(n) \wedge R(n) \wedge P(n) \wedge C(n) \wedge A(n) \wedge Q(n)$ holds, as required

By this inductive argument we established that the conjunction $S(n) \wedge R(n) \wedge P(n) \wedge C(n) \wedge A(n) \wedge Q(n)$ holds for all $n \geq 1$.

□

We now bring all the above results together to obtain the desired result that the rewriting procedure produces well-defined n-diagrams for any $n \geq 0$.

Theorem 2.3.28. *For any well-defined n-diagrams D, S, T such that S and T are globular with respect to each other, and a well-defined embedding $e : S \hookrightarrow D$, the rewrite $D.\Pi[e, T]$ is a well-defined diagram.*

Proof. Two separate cases follow by different results proved previously:

- For $n = 0$ follows by Lemma 2.3.17.
- For $n > 0$, by Theorem 2.3.27, we have that $R(n)$ holds for any $n \geq 1$

By this we established that the rewrite $D.\Pi[e, T]$ of a well-defined n-diagram D is also well-defined. □

Theorem 2.3.29. *Given well-defined n-diagrams such that S, T are globular with respect to each other and for a well-defined embedding $e : S \hookrightarrow A$, the lifted embedding $e.\Lambda[T] : T \hookrightarrow A.\Pi[e, T]$ is well-defined.*

Proof. This result follows immediately by Lemma 2.3.24, since by Theorem 2.3.28, we have that $R(n)$ is true for all $n \geq 0$. □

The final observation about the process of rewriting is that we may express it using higher level cells. The rewrite of an n-diagram D into $D.\Pi[e, T]$ gives rise to an $(n+1)$-diagram R of the following form:

$$R = \begin{cases} R.s = D \\ R[0].g = g & \text{such that } s(g) = S, t(g) = T \\ R[0].e = e \end{cases}$$

This observation is crucial for our higher dimensional rewriting perspective and universal treatment of $(n+1)$-cells as rewrites between composite n-cells and gives us a wider expressivity to reason about higher-level cells.

2.4 Composition

The next step towards a higher dimensional rewriting system and to modelling quasistrict n-categories is the ability to express composite objects and relations, so that they can then be subject to higher-level relations. In our scheme this is achieved through diagram *composition*. There are several requirements that need to be fulfilled in order for two diagrams to be composable. Firstly, they need to be built over the same signature σ, in this section we always assume that this is the case. Secondly, the target boundary and the source boundary along which we are composing must match, this is so that the diagrams can be 'glued' together. Crucially, the diagrams themselves do not need to by of the same dimension, but their boundaries do. Allowing diagrams of different dimensions to be composed is consistent with the whiskering approach to cell composition in higher categories.

In the initial part of this section we introduce some auxiliary notions that make defining the composite of two diagrams more dependent on the results on embedding composition that have already been proved in Section 2.3.

2.4.1 Identity embeddings

Definition 2.4.1. Given a well-defined n-diagram D, the *identity embedding* $\mathrm{id}_D : D \hookrightarrow D$ is defined as follows:

- $\mathrm{id}_D.h = 0$
- If $n > 0$, then also: $\mathrm{id}_D.e = \mathrm{id}_{D.s}$

The intuition here is that every diagram is trivially embedded in itself.

Lemma 2.4.2. *Given a well-defined n-diagram S and the identity embedding $\mathrm{id}_S : S \hookrightarrow S$ and a well-defined diagram T globular with respect to S, the following holds:*

$$\mathrm{id}_S.\Lambda[T] = \mathrm{id}_T$$

Proof. If $n = 0$, by Definition 2.3.6, to show that these embeddings are equivalent, we need to check that the domains and codomains of these embeddings are equivalent diagrams. The types of these embeddings are as follows:

$$\begin{aligned} &\mathrm{id}_T : T \hookrightarrow T && \text{[Def. 2.4.1]} \\ &\mathrm{id}_S.\Lambda[T] : T \hookrightarrow S.\Pi[\mathrm{id}_S, T] && \text{[Def. 2.3.1]} \end{aligned}$$

By applying Definition 2.3.4 we obtain that $S.\Pi[\mathrm{id}_S, T] = T$, hence the domains and codomains of both embeddings match.

If $n > 0$, we additionally need to check that:

- Component embeddings are equivalent:

$$(\mathrm{id}_S.\Lambda[T]).e = (\mathrm{id}_S).e \qquad \text{[Def. 2.3.1]}$$

- Heights are equal:

$$(\mathrm{id}_S.\Lambda[T]).h = (\mathrm{id}_S).h \qquad \text{[Def. 2.3.1]}$$

By this argument both embeddings are equivalent, as required. □

A desired property of an identity embedding is that composing it with any other diagram embedding e, should have no effect on e.

Lemma 2.4.3. *Given a well-defined n-diagram D, the identity embedding $\mathrm{id}_D : D \hookrightarrow D$ and a well-defined embeddings $e : A \hookrightarrow D$ and $f : D \hookrightarrow B$, the following holds:*

$$\begin{aligned} \mathrm{id}_D \circ e &= e \\ f \circ \mathrm{id}_D &= f \end{aligned}$$

Proof. Let us prove that $\mathrm{id}_D \circ e = e$ by induction on n.

- *Base case:* For $n = 0$, by Definition 2.3.6 domains and codomains of both embeddings must be equal. The types are as follows:

$$\begin{aligned} &e : A \hookrightarrow D \\ &\mathrm{id}_D \circ e : A \hookrightarrow D \end{aligned}$$

  Domains and codomains trivially are equivalent diagrams.

- *Inductive step:* For $n > 0$, we assume that the result holds for $(n-1)$-diagram embeddings *(IH)*. Types are the same as in the base case, we additionally need to check that:

  ⋆ Component embeddings are equivalent:

$$\begin{aligned} &(\mathrm{id}_D \circ e).e = \\ &\quad = (\mathrm{id}_D).e.\Lambda[D[e.h].d] \circ e.e && \text{[Def. 2.3.2]} \\ &\quad = (\mathrm{id}_{D[0].d}).\Lambda[D[e.h].d] \circ e.e && \text{[Def. 2.4.1]} \\ &\quad = \mathrm{id}_{D[e.h].d} \circ e.e && \text{[Def. 2.4.2]} \\ &\quad = e.e && [\mathit{IH}] \end{aligned}$$

* ⋆ Heights are equal:

$$\begin{aligned}
&(\mathrm{id}_D \circ e).h = \\
&\quad = (\mathrm{id}_D).h + e.h && \text{[Def. 2.3.2]} \\
&\quad = e.h && \text{[Def. 2.4.1]}
\end{aligned}$$

By this inductive argument both embeddings are equivalent for $n \geq 0$, as required. The proof of the other equivalence follows similarly. □

With the aid of these results we show that the identity embedding is well-defined.

Lemma 2.4.4. *Given a well-defined n-diagram D, the identity embedding $\mathrm{id}_D : D \hookrightarrow D$ is well-defined.*

Proof. id_D is an endomorphism of a well-defined diagram D, hence trivially both its domain and codomain are well-defined.

We prove well-definedness of this embedding by induction on n.

- *Base case:* For $n = 0$, by Definition 2.2.8 we just need the single cell in the domain being equal to the single cell in the codomain *i.e.* $D[0].g = D[0].g$ which holds.
- *Inductive step:* For $n > 0$, assume that all identity embeddings of $(n-1)$-diagrams are well-defined.

  By Definition 2.2.8 we need to check three conditions:

  * ⋆ The component embedding is well-defined. By Definition 2.4.1, we have $\mathrm{id}_D.e = \mathrm{id}_{D.s} = \mathrm{id}_{D[0].d}$. Since $D[0].d$ is an $(n-1)$-diagram, $\mathrm{id}_D.e$ is well-defined by the inductive hypothesis.
  * ⋆ Corresponding generators are equal. For $0 \leq i \leq |D|$ we have:

  $$D[i + (\mathrm{id}_D).h].g = D[i].g \qquad \text{[Def. 2.4.1]}$$

  * ⋆ Corresponding embeddings satisfy the following for $0 \leq i \leq |D|$:

  $$\begin{aligned}
  &((\mathrm{id}_D).e).\Lambda[D[i].d] \circ D[i].e \\
  &\quad = (\mathrm{id}_{D.s}).\Lambda[D[i].d] \circ D[i].e && \text{[Def. 2.4.1]} \\
  &\quad = (\mathrm{id}_{D[0].d}).\Lambda[D[i].d] \circ D[i].e \\
  &\quad = \mathrm{id}_{D[i].d} \circ D[i].e && \text{[Def. 2.4.2]} \\
  &\quad = D[i].e && \text{[Def. 2.4.3]}
  \end{aligned}$$

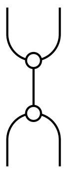
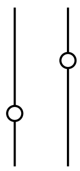
    As required.

  As all these conditions hold, id_D is well-defined.

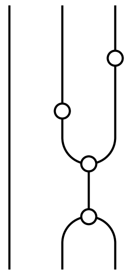
By this inductive argument we established that id_D is well-defined for any n-diagram D for all $n \geq 0$. □

2.4.2 Defining composition

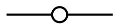
The next major operation modifying the diagram structure is diagram *composition*. Recall from Section 1.4 that intuitively this corresponds to glueing two diagrams together along a common boundary. For instance for the following diagrams S and D:

S =  D = 

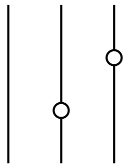
Their composite, which we denote by $S \circ D$ is:

$S \circ D \quad =$ 

Note that we can also compose diagrams that are of different dimension, for instance given a 1-diagram M:

$M \quad =$ 

The composite $M \circ D$ is:



To be able to compose two diagrams D and S, we require means to alter embeddings $e : X \hookrightarrow D$ to embed in the composite diagram instead, to that end we define two *inclusion* embeddings, the *right inclusion* $\mathrm{Inc}_r(S, D) : D \hookrightarrow S \circ D$ and the *left inclusion* $\mathrm{Inc}_l(S, D) : D \hookrightarrow D \circ S$. Recall that we write $s^k(D)$ and $t^k(D)$ to denote taking the source or the target of the diagram k times, as explained in the discussion of Definition 2.2.5.

Definition 2.4.5. Given an n-diagram D and an m-diagram S:

- If $n \geq m$ and $t(S) = s^{n-m+1}(D)$ and the composite $S \circ D$ exists, we define the *inclusion* embedding $\mathrm{Inc}_r(S, D) : D \hookrightarrow S \circ D$ in the following way:

  ⋆ If $n = m$, then:

$$\mathrm{Inc}_r(S, D).h = |S|$$
$$\mathrm{Inc}_r(S, D).e = \mathrm{id}_D.e$$

  ⋆ If $n > m$, then:

$$\mathrm{Inc}_r(S, D).h = \mathrm{id}_D.h$$
$$\mathrm{Inc}_r(S, D).e = \mathrm{Inc}_r(S, D.s)$$

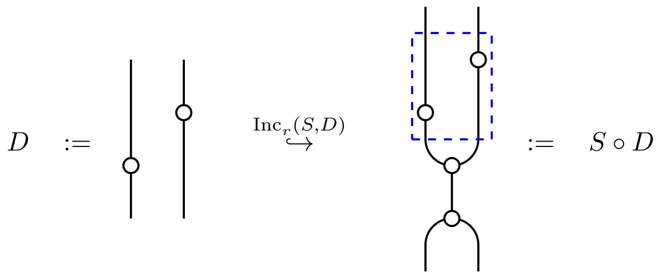
- If $n < m$ and $t^{m-n+1}(S) = s(D)$ and the composite $S \circ D$ exists, we define the inclusion embedding $\mathrm{Inc}_l(S, D) : S \hookrightarrow S \circ D$ in the following way:

$$\mathrm{Inc}_l(S, D).h = \mathrm{id}_S.h$$
$$\mathrm{Inc}_l(S, D).e = \mathrm{Inc}_r(S.s, D)$$

Note that, in the latter case, the composition of D with S has no effect on the embedding data. However we do need to change the embedding codomain, so that the types match.

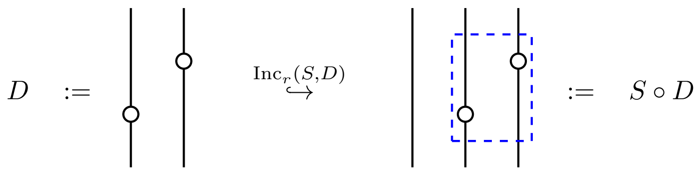
The intuition behind the inclusion embeddings is that a diagram D is automatically a subdiagram of the composite $S \circ D$, the inclusion embedding witnesses this relation. There is a need for two variations, left and right inclusions, because D can be composed with another diagram either on the left to form

$D \circ S$, or on the right to form $S \circ D$. Let us illustrate this with two examples, first consider the following diagrams S and D that are of the same dimension:



In this picture, we indicate the inclusion embedding $\mathrm{Inc}_r(S, D)$ by placing a dashed rectangle over the instance of D appearing in $S \circ D$.

Now, consider S and D such that the dimension of D is larger:



Here again, we indicate $\mathrm{Inc}_r(S, D)$ by the dashed rectangle.

In this setup, an n-diagram D and an m-diagram S can be composed in exactly one way, along their common $\max(m, n) - min(m, n) + 1$ boundary. In particular, two n-diagrams can only be composed along their matching target and source, this corresponds to vertical composition. We believe that this approach is still fully general and permits construction of any arbitrary n-diagram, additionally it helps in easier resolution of ambiguities that arise if other methods of composition are allowed. As stated before, this perspective is consistent with the approach to cell composition through whiskering. An example how horizontal composition of 2-cells in a 2-category may be retrieved is given in Section 1.4.

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 as does $n - m$. Eventually, we decrease n sufficiently, so that $n = m$ and then the recursion bottoms out with the base clause. The case for $m > n$ is analogous. This ensures that the definition is well-founded.

Definition 2.4.6. Given an n-diagram D and an m-diagram S such that $t(S) = s^{n-m+1}(D)$ if $m \leq n$ or $t^{m-n+1}(S) = s(D)$ otherwise, then the *composite* diagram $S \circ_{m,n} D$ is defined as follows:

- If $n = m$, the individual components of $S \circ_{m,n} D$ are as follows:

$$(S \circ_{m,n} D).s = S.s \tag{2.9}$$

$$|S \circ_{m,n} D| = |D| + |S| \tag{2.10}$$

$$(S \circ_{m,n} D)[j].g = \begin{cases} S[j].g & \text{for } 0 \leq j < |S| \\ D[j - |S|].g & \text{for } |S| \leq j < |D| + |S| \end{cases} \tag{2.11}$$

$$(S \circ_{m,n} D)[j].e = \begin{cases} S[j].e & \text{for } 0 \leq j < |S| \\ D[j - |S|].e & \text{for } |S| \leq j < |D| + |S| \end{cases} \tag{2.12}$$

- If $m < n$, the individual components of $S \circ_{m,n} D$ are as follows:

$$(S \circ_{m,n} D).s = S \circ_{m,n-1} D.s \tag{2.13}$$
$$|S \circ_{m,n} D)| = |D| \tag{2.14}$$
$$(S \circ_{m,n} D)[j].g = D[j].g \quad \text{for } 0 \leq j < |D| \tag{2.15}$$
$$(S \circ_{m,n} D)[j].e = \text{Inc}_r(S, D[j].d) \circ D[j].d \quad \text{for } 0 \leq j < |D| \tag{2.16}$$

- If $n < m$, the individual components of $S \circ_{m,n} D$ are as follows:

$$(S \circ_{m,n} D).s = S.s \circ_{m-1,n} D \tag{2.17}$$
$$|S \circ_{m,n} D| = |S| \tag{2.18}$$
$$(S \circ_{m,n} D)[j].g = S[j].g \quad \text{for } 0 \leq j < |S| \tag{2.19}$$
$$(S \circ_{m,n} D)[j].e = \text{Inc}_l(S[j].d, D) \circ S[i].e \quad \text{for } 0 \leq j < |S| \tag{2.20}$$

As stated before, the type of composition is uniquely determined by the dimensions of the diagrams being composed. As a shorthand for a composite of an n-diagram D and an m-diagram S, we label $S \circ_{m,n} D$ by $a = min(n, m) - 1$ instead. That way the terminology more closely matches the naming scheme for composition which is standard in category theory. This leads to a certain overloading of notation since we use the same label for composites $S \circ aD$ and $D \circ_a S$, however since these cases behave analogously, this should not cause any confusion. Furthermore, if the dimensions are completely unambiguous, we omit the subscripts entirely and simply say $D \circ S$ to refer to the composite diagram.

For composition of two n-diagrams S, D, intuitively we just concatenate their lists of generators and embeddings. Due to that, the lists of slices also get concatenated. This is formalised by the following lemma.

Lemma 2.4.7. *Given well-defined n-diagrams S, D, such that $t(S) = s(D)$, the following holds for slices of the composed diagram for $0 \leq j \leq |S \circ D|$:*

$$(S \circ D)[j].d = \begin{cases} S[j].d & 0 \leq j \leq |S| \\ D[j - |S|].d & |S| \leq j \leq |S| + |D| \end{cases}$$

Proof. Note that, by 2.10, we obtain that: $|S \circ D| = |S| + |D|$, so let $0 \leq j \leq |S| + |D|$ and consider two ranges separately:

- For $0 \leq j \leq |S|$ we show the result by induction on j:
  - ⋆ *Base case:* For $j = 0$, we have the following:

$$\begin{aligned} &(S \circ D)[0].d \\ &\quad = (S \circ D).s && \text{[Def. 2.2.5]} \\ &\quad = S.s && \text{[Eq. (2.9)]} \\ &\quad = S[0].d && \text{[Def. 2.2.5]} \end{aligned}$$

  - ⋆ *Inductive step:* For $0 < j \leq |S|$, we assume that $(S \circ D)[j].d = S[j].d.$ Let us consider $(S \circ D)[j + 1].d$:

$$\begin{aligned} &(S \circ D)[j + 1].d \\ &\quad = (S \circ D)[j].d.\Pi[(S \circ D)[j].e, t((S \circ D)[j].g)] && \text{[Def. 2.2.3]} \\ &\quad = (S \circ D)[j].d.\Pi[(S \circ D)[j].e, t(S[j].g)] && \text{[Eq. (2.11)]} \\ &\quad = (S \circ D)[j].d.\Pi[S[j].e, t(S[j].g)] && \text{[Eq. (2.12)]} \\ &\quad = S[j].d.\Pi[S[j].e, t(S[j].g)] && [IH] \\ &\quad = S[j + 1].d && \text{[Def. 2.2.3]} \end{aligned}$$

- For $|S| \leq j \leq |S| + |D|$, note that by the argument for $0 \leq j \leq |S|$ we obtain:

$$
\begin{aligned}
&(S \circ D)[|S|].d \\
&\quad = S[|S|].d \\
&\quad = t(S) && \text{[Def. 2.2.5]} \\
&\quad = s(D) && \text{[assumption]} \\
&\quad = D[0].d && \text{[Def. 2.2.5]}
\end{aligned}
$$

  The result for $|S| \leq j \leq |S| + |D|$ is proved inductively in a similar way as the result for $0 \leq j \leq |S|$.

By these two inductive arguments the result is shown for $0 \leq j \leq |S \circ D|$.

□

2.4.3 Correctness of the composition construction

We use the same proof technique as for rewriting. Again, due to the heavily recursive nature of the diagram and signature structures and their mutual references, we introduce several logical statements about properties of composite diagrams and inclusion embeddings. For two diagrams being composed: an n-diagram D and an m-diagram S the statements depend on an integer $k \geq 0$ such that $k = |n - m|$. That way, the base case in our recursion covers the situation when the two diagrams composed are of the same dimension. Again, there is the main inductive proof that the conjunction of logical statements $K(k), L(k), M(k), N(k)$ holds for all $k \geq 0$. Since we make no other assumptions about m, n this ensures that we show the results for any combination of dimensions. Again, the overall structure is that we first present the statements, then we show that they hold for $k = 0$ to establish base cases. This is followed by proofs of a series of implications and finally, in the conclusion of this section, all the lemmas are brought together to establish the main result on the composite of two diagrams being well-defined.

Below, we define four logical statements on properties of diagram composition and inclusion embeddings:

Definition 2.4.8 ($L(k)$)**.** For $k \geq 0$, let $L(k)$ denote the statement that for any well-defined n-diagram D and a well-defined m-diagram S such that $|n - m| = k$ and $t(S) = s^{n-m+1}(D)$ if $m \leq n$ or $t^{m-n+1}(S) = s(D)$ otherwise, then the *composite* diagram $S \circ D$ is well-defined.

The inclusion embedding which is necessary to define a composite of two diagrams necessarily has to be well-defined.

Definition 2.4.9 ($N(k)$)**.** For $k \geq 0$, let $N(k)$ denote the statement that for any well-defined n-diagram D and a well-defined diagram m-diagram S such that $|n - m| = k$:

- If $n \geq m$ and $t(S) = s^{n-m+1}(D)$ and the composite $S \circ D$ exists, the inclusion embedding $\mathrm{Inc}_r(S, D) : D \hookrightarrow S \circ D$ is well-defined.
- If $n < m$ and $t^{m-n+1}(S) = s(D)$ and the composite $S \circ D$ exists, the inclusion embedding $\mathrm{Inc}_l(S, D) : D \hookrightarrow D \circ S$ is well-defined.

A slice of the composite diagram can be written as a composite of the diagram of lower dimension and the appropriate slice of the diagram of higher dimension.

Definition 2.4.10 ($K(k)$)**.** For $k \geq 1$, let $K(k)$ denote the statement that for any n-diagram D and any m-diagram S such that $|n - m| = k$ and such that the composite $S \circ D$ exists, the following equalities hold:

If $n > m$	$(S \circ D)[i].d = S \circ (D[i].d)$	for any $0 \leq i < \vert D\vert$
If $n < m$	$(S \circ D)[i].d = (S[i].d) \circ D$	for any $0 \leq i < \vert S\vert$

The inclusion embedding into a slice of the composite diagram, can instead be expressed using the lifted embedding.

Definition 2.4.11 ($M(k)$)**.** For $k \geq 1$, let $M(k)$ denote the statement that for any well-defined n-diagram D and a well-defined diagram m-diagram S such that $|n - m| = k$ for any $0 \leq i < |D|$ the following equality holds:

$$\begin{array}{lll} \text{If } n > m & \mathrm{Inc}_r(S, D[i].d) = (\mathrm{Inc}_r(S, D).e).\Lambda[D[i].d] & \text{for any } 0 \leq i < |D| \\ \text{If } n < m & \mathrm{Inc}_l(S[i].d, D) = (\mathrm{Inc}_l(S, D).e).\Lambda[S[i].d] & \text{for any } 0 \leq i < |S| \end{array}$$

We now proceed to prove several lemmas that establish base cases for the main recursive proof:

Lemma 2.4.12. *For $k = 0$ the following holds: $L(0) \implies N(0)$*

Proof. We assume that L(0) holds, *i.e.* any two well-defined n-diagrams D, S such that $t(S) = s(D)$, the *composite* diagram $S \circ D$ is well-defined.

As $k = 0$, implies $n = m$, we only need to show that the inclusion embedding $\mathrm{Inc}_r(S, D) : D \hookrightarrow S \circ D$ is well-defined for $N(0)$ to hold.

The domain D of the embedding $\mathrm{Inc}_r(S, D)$ is well-defined by assumption. The codomain $S \circ D$ is well-defined by $L(0)$. Since $n, m \geq 1$, by Definition 2.2.8 $\mathrm{Inc}_r(S, D)$ needs to satisfy three separate conditions:

- The component embedding $\mathrm{Inc}_r(S, D).e$ is well-defined. As $n = m$, by Definition 2.4.5, $\mathrm{Inc}_r(S, D).e = \mathrm{id}_D.e = \mathrm{id}_{D.s}$, which is well-defined by lemma2.4.4.

- For every $0 \leq j < |D|$ we have:

$$\begin{aligned} &(S \circ D)[j + \mathrm{Inc}_r(S, D).h].g \\ &\quad = (S \circ D)[j + |S|].g && \text{[Def. 2.4.5]} \\ &\quad = D[(j + |S|) - |S|].g && \text{[Eq. (2.11)]} \\ &\quad = D[j].g \end{aligned}$$

  As required.

- For every $0 \leq j < |D|$ we have:

$$\begin{aligned} &\mathrm{Inc}_r(S, D).e.\Lambda[D[j].d] \circ D[j].e \\ &\quad = \mathrm{id}_D.e.\Lambda[D[j].d] \circ D[j].e && \text{[Def. 2.4.5]} \\ &\quad = \mathrm{id}_{D.s}.\Lambda[D[j].d] \circ D[j].e && \text{[Def. 2.4.1]} \\ &\quad = \mathrm{id}_{D[0].d}.\Lambda[D[j].d] \circ D[j].e && \text{[Def. 2.2.5]} \\ &\quad = \mathrm{id}_{D[j].d} \circ D[j].e && \text{[Def. 2.4.2]} \\ &\quad = D[j].e && \text{[Def. 2.4.3]} \\ &\quad = D[(j + |S|) - |S|].e \\ &\quad = (S \circ D)[j + |S|].e && \text{[Eq. (2.11)]} \\ &\quad = (S \circ D)[j + \mathrm{Inc}_r(S, D).h].e && \text{[Def. 2.4.5]} \end{aligned}$$

  As required.

As all three conditions are satisfied, we conclude that for any two well-defined n-diagrams D, S such that $t(S) = s(D)$, the inclusion embedding $\mathrm{Inc}_r(S, D)$ is well-defined. ☐

Lemma 2.4.13. *The statement $L(0)$ holds without any further assumptions.*

Proof. For $k = 0$, we have $n = m$, so to establish $L(0)$ we need to show that given two well-defined n-diagrams S, D such that $t(S) = s(D)$, the composite $S \circ D$ is well-defined.

Since $m, n \geq 1$, by Definition 2.2.6, the diagram $S \circ D$ is well-defined if the following two conditions are satisfied:

- The source $(S \circ D).s = (S \circ D)[0].d$ is a well-defined diagram.
- For every $0 < j \leq |S \circ D|$ the slice $(S \circ D)[j].d$ exists and is well-defined.

As $n = m$, we apply Lemma 2.4.7 to obtain that:

$$(S \circ D)[j].d = \begin{cases} S[j].d & 0 \leq j \leq |S| \\ D[j - |S|].d & |S| \leq j \leq |S| + |D| \end{cases}$$

As both S, D are well-defined diagrams, then all their slices are also well-defined. Since every slice of $S \circ D$ is equal to either a slice of S or a slice of D, they are all well-defined, hence $S \circ D$ is also well-defined. □

Lemma 2.4.14. *The statement $K(1)$ holds without any further assumptions.*

Proof. We need to show that for any n-diagram D and m-diagram S such that the composite $S \circ D$ exists and

$$\begin{array}{lll} \text{If } n > m & (S \circ D)[i].d = S \circ (D[i].d) & \text{for any } 0 \leq i < |D| \\ \text{If } n < m & (S \circ D)[i].d = (S[i].d) \circ D & \text{for any } 0 \leq i < |S| \end{array}$$

We consider both these cases separately, first let $n > m$. We prove the result by induction on $0 \leq j \leq |D|$.

- *Base case:* For $j = 0$, the result follows immediately form the definitions:

$$\begin{aligned} &(S \circ D)[0].d = \\ &\qquad = (S \circ D).s && \text{[Def. 2.2.5]} \\ &\qquad = S \circ (D.s) && \text{[Def. 2.4.6]} \\ &\qquad = S \circ (D[0].d) && \text{[Def. 2.2.5]} \end{aligned}$$

- *Inductive step:* For $j > 0$, assume that:

$$(S \circ D)[j].d = S \circ (D[j].d) \qquad (IH)$$

  Let us consider $(S \circ D)[j + 1].d$ and $S \circ (D[j + 1].d)$, to show that two diagrams are equivalent, by Definition 2.3.7, we need to check the following:

  - ⋆ Sources are equivalent diagrams:

$$\begin{aligned} &((S \circ D)[j + 1].d).s \\ &\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, t((S \circ D)[j].g)]).s && \text{[Def. 2.2.3]} \\ &\quad = ((S \circ D)[j].d).s && \text{[Eq. (2.5)]} \\ &\quad = (S \circ (D[j].d)).s && [IH] \\ &\quad = S.s && \text{[Eq. (2.9)]} \\ &\quad = (S \circ (D[j + 1].d)).s && \text{[Eq. (2.9)]} \end{aligned}$$

  - ⋆ Lengths of generator lists are equal:
    Note that even though $m = n-1$, we still have $m < n$, hence the following is true for generators of the composed diagram for $0 \leq j < |D|$:

$$(S \circ D)[j].g = D[j].g \qquad \text{[Def. 2.15]}$$

The comparison of lengths for the two diagrams is as follows:

$$
\begin{aligned}
&|(S \circ D)[j+1].d| \\
&\quad = |(S \circ D)[j].d).\Pi[(S \circ D)[j].e, t((S \circ D)[j].g))| && \text{[Def. 2.2.3]} \\
&\quad = |S \circ D)[j].d| - |s((S \circ D)[j].g)| + |t((S \circ D)[j].g)| && \text{[Eq. (2.6)]} \\
&\quad = |S \circ (D[j].d)| - |s((S \circ D)[j].g)| + |t((S \circ D)[j].g)| && [IH] \\
&\quad = |S \circ (D[j].d)| - |s(D[j].g| + |t(D[j].g| && \text{[Eq. (2.15)]} \\
&\quad = |S| + |D[j].d| - |s(D[j].g| + |t(D[j].g| && \text{[Eq. (2.10)]} \\
&\quad = |S| + |D[j+1].d| && \text{[Eq. (2.6)]} \\
&\quad = |S \circ (D[j+1].d)| && \text{[Eq. (2.10)]}
\end{aligned}
$$

⋆ For generators and embeddings we need to show that for $0 \leq k < |(S \circ D)[j+1].d| = |S \circ (D[j+1].d)|$, the k-th generators in generator lists of both diagrams correspond and the same for k-th embeddings.

Since $m = n - 1$, we can simplify the height of the j-th embedding in $S \circ D$ in the following way.

$$
\begin{aligned}
&((S \circ D)[j].e).h = \\
&\quad = (\mathrm{Inc}_r(S, D[j].d) \circ D[j].e).h && \text{[Eq. (2.12)]} \\
&\quad = (\mathrm{Inc}_r(S, D[j].d)).h + D[j].e.h && \text{[Def. 2.3.2]} \\
&\quad = |S| + D[j].e.h && \text{[Def. 2.4.5]}
\end{aligned}
$$

Let us refer to this equality as [∗].

We show the necessary equivalences separately for four individual ranges:

∗ In the range: $0 \leq k < |S|$, we have:
Generators:

$$
\begin{aligned}
&((S \circ D)[j+1].d)[k].g \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\quad t((S \circ D)[j].g)]).g && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d)[k].g && \text{[Eq. (2.7)]} \\
&\quad = (S \circ (D[j].d))[k].g && [IH] \\
&\quad = S[k].g && \text{[Eq. (2.11)}, k < |S|] \\
&\quad = (S \circ (D[j+1].d))[k].g && \text{[Eq. (2.11)}, k < |S|]
\end{aligned}
$$

Embeddings:

$$
\begin{aligned}
&((S \circ D)[j+1].d)[k].e \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\quad t((S \circ D)[j].g)])[k].e && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d)[k].e && \text{[Eq. (2.8)]} \\
&\quad = (S \circ (D[j].d))[k].e && [IH] \\
&\quad = S[k].e && \text{[Eq. (2.12)}, k < |S|] \\
&\quad = (S \circ (D[j+1].d))[k].e && \text{[Eq. (2.11)}, k < |S|]
\end{aligned}
$$

∗ In the range: $|S| \leq k < (S \circ D)[j].e.h$
We have: $(S \circ D)[j].e = D[i - |S|].e$, hence the range is equivalent to:

$$|S| \leq k < D[i - |S|].e.h$$

This means that when we apply Definition 2.3.4, we use generators and embeddings from the original diagram, not from the target of the rewrite.
Generators:

$$\begin{aligned}
&((S \circ D)[j+1].d)[k].g \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\quad t((S \circ D)[j].g)]).g && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d)[k].g && \text{[Eq. (2.7)]} \\
&\quad = (S \circ (D[j].d))[k].g && [IH] \\
&\quad = (D[j].d)[k - |S|].g && \text{[Eq. (2.11)}, |S| \leq k] \\
&\quad = (D[j].d.\Pi[D[j].d, t(D[j].g)])[k - |S|].g && \text{[Eq. (2.7)]} \\
&\quad = (D[j+1])[(k - |S|].g && \text{[Def. 2.2.3]} \\
&\quad = (S \circ (D[j+1].d))[k].g && \text{[Eq. (2.11)}, |S| \leq k]
\end{aligned}$$

Embeddings:

$$\begin{aligned}
&((S \circ D)[j+1].d)[k].e \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\quad t((S \circ D)[j].g)]).e && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d)[k].e && \text{[Eq. (2.8)]} \\
&\quad = (S \circ (D[j].d))[k].e && [IH] \\
&\quad = (D[j].d)[k - |S|].e && \text{[Eq. (2.12)}, |S| \leq k] \\
&\quad = (D[j].d.\Pi[D[j].d, t(D[j].g)])[k - |S|].e && \text{[Eq. (2.8)]} \\
&\quad = (D[j+1])[(k - |S|].e && \text{[Def. 2.2.3]} \\
&\quad = (S \circ (D[j+1].d))[k].e && \text{[Eq. (2.12)}, |S| \leq k]
\end{aligned}$$

∗ In the range: $(S \circ D)[j].e.h \leq k < (S \circ D)[j].e.h + |t((S \circ D)[j].g)|$
We have: $(S \circ D)[j].g = D[j].g$, hence the range is equivalent to:

$$|S| + D[j].e.h \leq k < |S| + D[j].e.h + |t(D[j].g)|$$

Generators:

$$\begin{aligned}
&((S \circ D)[j+1].d)[k].g \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\quad t((S \circ D)[j].g)]).g && \text{[Def. 2.2.3]} \\
&\quad = t((S \circ D)[j].g)[k - (S \circ D)[j].e.h] && \text{[Eq. (2.7)]} \\
&\quad = t(D[j].g)[k - (S \circ D)[j].e.h].g && \text{[Eq. (2.15)]} \\
&\quad = t(D[j].g)[(k - (D[j].e.h + |S|)].g && [*] \\
&\quad = t(D[j].g)[(k - |S|) - D[j].e.h].g \\
&\quad = (D[j+1].d)[(k - |S|].g && \text{[Eq. (2.7)]} \\
&\quad = (S \circ (D[j+1].d))[k].g && \text{[Eq. (2.11)}, |S| \leq k]
\end{aligned}$$

Embeddings:

$$
\begin{aligned}
&((S \circ D)[j+1].d)[k].e \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, t((S \circ D)[j].g)]).e && \text{[Def. 2.2.3]} \\
&\quad = (((S \circ D)[j].e).e).\Lambda[t((S \circ D)[j].g)[k - (S \circ D)[j].e.h].d] \\
&\qquad\quad \circ t((S \circ D)[j].g)[k - (S \circ D)[j].e.h].e && \text{[Eq. (2.7)]} \\
&\quad = ((\mathrm{Inc}_r(S, D[j].d) \circ D[j].e).e).\Lambda[\\
&\qquad\qquad t((S \circ D)[j].g)[k - (S \circ D)[j].e.h].d] \\
&\qquad\qquad\qquad \circ t((S \circ D)[j].g)[k - (S \circ D)[j].e.h].e && \text{[Eq. (2.16)]} \\
&\quad = ((\mathrm{Inc}_r(S, D[j].d) \circ D[j].e).e).\Lambda[t(D[j].g)[k - (S \circ D)[j].e.h].d] \\
&\qquad\quad \circ t(D[j].g)[k - (S \circ D)[j].e.h].e && \text{[Eq. (2.15)]} \\
&\quad = ((\mathrm{Inc}_r(S, D[j].d) \circ D[j].e).e).\Lambda[t(D[j].g)[k - (|S| + D[j].e.h)] \\
&\qquad\quad \circ t(D[j].g)[k - (|S| + D[j].e.h)].e && [*] \\
&\quad = ((\mathrm{Inc}_r(S, D[j].d).e).\Lambda[(D[j].d)[D[j].e.h].d] \\
&\qquad\quad \circ D[j].e.e).\Lambda[t(D[j].g)[k - (|S| + D[j].e.h)] \\
&\qquad\qquad\qquad \circ t(D[j].g)[k - (|S| + D[j].e.h)].e && \text{[Def. 2.3.2]} \\
&\quad = (((\mathrm{id}_{D[j].d}).e).\Lambda[(D[j].d)[D[j].e.h].d] \\
&\qquad\quad \circ D[j].e.e).\Lambda[t(D[j].g)[k - (|S| + D[j].e.h)] \\
&\qquad\qquad\qquad \circ t(D[j].g)[k - (|S| + D[j].e.h)].e && \text{[Def. 2.4.5]} \\
&\quad = ((\mathrm{id}_{(D[j].d)[0].d}).\Lambda[(D[j].d)[D[j].e.h].d] \\
&\qquad\quad \circ D[j].e.e).\Lambda[t(D[j].g)[k - (|S| + D[j].e.h)] \\
&\qquad\qquad\qquad \circ t(D[j].g)[k - (|S| + D[j].e.h)].e && \text{[Def. 2.4.1]} \\
&\quad = (\mathrm{id}_{(D[j].d)[D[j].e.h].d}) \\
&\qquad\quad \circ D[j].e.e).\Lambda[t(D[j].g)[k - (|S| + D[j].e.h)] \\
&\qquad\qquad\qquad \circ t(D[j].g)[k - (|S| + D[j].e.h)].e && \text{[Def. 2.4.2]} \\
&\quad = D[j].e.e).\Lambda[t(D[j].g)[k - (|S| + D[j].e.h)] \\
&\qquad\quad \circ t(D[j].g)[k - (|S| + D[j].e.h)].e && \text{[Def. 2.4.3]} \\
&\quad = (D[j].e.e).\Lambda[t(D[j].g)[(k - |S|) - D[j].e.h].d] \\
&\qquad\quad \circ t(D[j].g)[(k - |S|) - D[j].e.h].e \\
&\quad = ((D[j].d).\Pi[D[j].e, t(D[j].g)])[k - |S|].e \\
&\quad = (D[j+1].d)[k - |S|].e && \text{[Eq. (2.7)]} \\
&\quad = (S \circ (D[j+1].d))[k].e && \text{[Eq. (2.11)]}
\end{aligned}
$$

$*$ In the range:

$$
\begin{aligned}
&(S \circ D)[j].e.h + |t((S \circ D)[j].g)| \leq k \\
&\qquad < |(S \circ D)[j].d| - |s((S \circ D)[j].g)| + |t((S \circ D)[j].g)|
\end{aligned}
$$

We have: $(S \circ D)[j].e = D[j - |S|].e$, hence the range is equivalent to:

$$
\begin{aligned}
&D[j - |S|].e.h + |t(D[j - |S|].g)| \leq k \\
&\qquad < |S| + |D[j - |S|]| - |s(D[j - |S|].g)| + |t(D[-|S|j].g)|
\end{aligned}
$$

Generators:

$$
\begin{aligned}
&((S \circ D)[j+1].d)[k].g \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\quad t((S \circ D)[j].g)]).g && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d)[k - |t((S \circ D)[j].g)| \\
&\qquad\quad + |s((S \circ D)[j].g)|].g && \text{[Eq. (2.7)]} \\
&\quad = (S \circ (D[j].d))[k - |t((S \circ D)[j].g)| \\
&\qquad\quad + |s((S \circ D)[j].g)|].g && [IH] \\
&\quad = (S \circ (D[j].d))[(k - |t(D[k].g)| + |s(D[k].g)|)].g && \text{[Eq. (2.15)}, |S| \leq k] \\
&\quad = (D[j].d)[(k - |t(D[k].g)| + |s(D[k].g)|) - |S|].g && \text{[Eq. (2.11)]} \\
&\quad = (D[j].d)[(k - |S|) - |t(D[k].g)| + |s(D[k].g)|].g \\
&\quad = (D[j].d.\Pi[D[j].e, t(D[j].g)])[k - |S|].e && \text{[Eq. (2.7)]} \\
&\quad = (D[j+1].d)[(k - |S|].g && \text{[Def. 2.2.3]} \\
&\quad = (S \circ (D[j+1].d))[k].g && \text{[Eq. (2.15)}, |S| \leq k]
\end{aligned}
$$

Embeddings:

$$
\begin{aligned}
&((S \circ D)[j+1].d)[k].e \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\quad t((S \circ D)[j].g)]).e && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d)[k - |t((S \circ D)[j].g)| \\
&\qquad\quad + |s((S \circ D)[j].g)|].e && \text{Eq. (2.8)]} \\
&\quad = (S \circ (D[j].d))[k - |t((S \circ D)[j].g)| \\
&\qquad\quad + |s((S \circ D)[j].g)|].e && [IH] \\
&\quad = (S \circ (D[j].d))[(k - |t(D[k].g)| + |s(D[k].g)|)].e && \text{[Eq. (2.12)}, |S| \leq k] \\
&\quad = (D[j].d)[(k - |t(D[k].g)| + |s(D[k].g)|) - |S|].e && \text{[Eq. (2.12)]} \\
&\quad = (D[j].d)[(k - |S|) - |t(D[k].g)| + |s(D[k].g)|].e \\
&\quad = (D[j].d).\Pi[D[j].e, t(D[j].g)])[k - |S|].e && \text{[Eq. (2.8)]} \\
&\quad = (D[j+1].d)[(k - |S|].g && \text{[Def. 2.2.3]} \\
&\quad = (S \circ (D[j+1].d))[k].e && \text{[Eq. (2.12)}, |S| \leq k]
\end{aligned}
$$

All embeddings and generators in both diagrams correspond. Hence, as all these conditions are satisfied, the two diagrams are equivalent by Definition 2.3.7. The argument for $n < m$ is analogous. By this we established that $K(1)$ holds.

□

With all the base cases established, we prove a series of implications between the logical statements defined earlier in this section. Again, for each implication we only take the minimal subset of expressions that implies the given statement for n.

Lemma 2.4.15. *For $k \geq 1$ the following holds: $N(k-1) \wedge L(k-1) \implies L(k)$.*

Proof. We assume that $N(k-1)$ holds, *i.e.* that for any well-defined x-diagram B and a well-defined y-diagram A, such that $|x - y| = k - 1$ and $t(A) = s^{x-y+1}(B)$, their composite $A \circ B$ is well-defined. We also assume that $L(k-1)$ holds.

Now consider two well-defined diagrams: an n-diagram D and an m-diagram S, such that $t(S) = s^{n-m+1}(D)$ and $|n - m| = k$. We need to show that $S \circ D$ is well-defined.

We consider two cases separately, first let $m \geq n$. By Definition 2.2.6, $S \circ D$ is well-defined if for $0 \leq j \leq |S \circ D|$ all the slices $(S \circ D)[j].d$ are well-defined.

We prove this result by induction on $0 \leq j \leq |S \circ D|$:

- *Base case:* For $j = 0$, we need to show that the source $(S \circ D)[0].d = (S \circ D).s$ is a well-defined diagram. As $m \leq n$, by the clause Eq. (2.13) in Definition 2.4.6 we obtain the following:

$$(S \circ D).s = S \circ (D.s)$$

  The dimension of $D.s$ is $n - 1$, hence we get $(n-1) - m = k - 1$ and since $L(k-1)$ holds we obtain that $(S \circ D).s$ is well-defined as the composite of two well-defined diagrams whose difference in dimensions is $k - 1$.

- *Inductive step:* For $0 < j \leq |S \circ D|$, we assume that the slice $(S \circ D)[j].d$ exists and is well-defined. Let us consider the subsequent slice $(S \circ D)[j+1].d$, then we have the following:

$$\begin{aligned}
&(S \circ D)[j+1].d = \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, t((S \circ D)[j].g)] && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, t(D[j].g)] && \text{[Eq. (2.15)]} \\
&\quad = ((S \circ D)[j].d).\Pi[\mathrm{Inc}_r(S, D[j].d) \circ D[j].e, t(D[j].g)] && \text{[Eq. (2.16)]}
\end{aligned}$$

  The following hold:

  - ⋆ $(S \circ D)[j].d$ is well-defined by the inductive hypothesis.
  - ⋆ $s(D[j].g)$ and $t(D[j].g)$ are globular with respect to each other by Definition 2.2.1.
  - ⋆ $D[j].e$ is well-defined, since D is well-defined.
  - ⋆ $\mathrm{Inc}_r(S, D[j].d)$ is well-defined, by application of $N(k-1)$, since the dimension of $D[j].d$ is $n - 1$.
  - ⋆ $\mathrm{Inc}_r(S, D[j].d) \circ D[j].e$ is well-defined as the composite of two well-defined embeddings by $C(n)$ 2.3.14 which holds by Theorem 2.3.27.

  Hence, we apply Theorem 2.3.28 to conclude that $(S \circ D)[j+1].d$ is a well-defined diagram as the rewrite of a well-defined diagram $(S \circ D)[j].d$.

By this inductive argument all slices $(S \circ D)[j].d$ are well-defined for $0 \leq j \leq |S \circ D|$, hence $S \circ D$ is well-defined.

The proof for $m > n$ is analogous. This establishes that $L(k)$ holds and the implication is true. □

Lemma 2.4.16. *For $k \geq 1$ the following holds: $L(k) \wedge M(k) \wedge N(k-1) \implies N(k)$.*

Proof. We make three assumptions:

- L(k) holds, *i.e.* that for any well-defined n-diagram D and a well-defined m-diagram S such that $|n - m| = k$ and $t(S) = s^{n-m+1}(D)$ if $m \leq n$ or $t^{m-n+1}(S) = s(D)$ otherwise, then the *composite* diagram $S \circ D$ is well-defined.

- $M(k)$ holds, *i.e.* that for any well-defined n-diagram D and a well-defined diagram m-diagram S such that $|n - m| = k$ the following equalities hold:

$$\begin{aligned}
&\text{If } n > m && \mathrm{Inc}_r(S, D[i].d) = (\mathrm{Inc}_r(S, D).e).\Lambda[D[i].d] && \text{for any } 0 \leq i < |D| \\
&\text{If } n < m && \mathrm{Inc}_l(S[i].d, D) = (\mathrm{Inc}_l(S, D).e).\Lambda[S[i].d] && \text{for any } 0 \leq i < |S|
\end{aligned}$$

- $N(k-1)$ holds.

We need to show that given an n-diagram D and an m-diagram S, such that $1 \leq m, n$:

- If $n > m$ and $t(S) = s^{n-m+1}(D)$ and the composite $S \circ D$ exists, the inclusion embedding $\mathrm{Inc}_r(S, D) : D \hookrightarrow S \circ D$ is well-defined.
- If $n < m$ and $t^{m-n+1}(S) \equiv s(D)$ and the composite $S \circ D$ exists, the inclusion embedding $\mathrm{Inc}_l(S, D) : D \hookrightarrow D \circ S$ is well-defined.

The case for $n = m$ cannot happen as $|n - m| = k \geq 1$. We consider the two above cases separately, first let $n > m$ and consider $\mathrm{Inc}_r(S, D)$:

The domain D of the embedding $\mathrm{Inc}_r(S, D)$ is well-defined by assumption. The codomain $S \circ D$ is well-defined by $L(k)$. By Definition 2.2.8 the inclusion embedding $\mathrm{Inc}_r(S, D) : D \hookrightarrow S \circ D$ is well-defined if the following three conditions are satisfied:

- The component embedding $\mathrm{Inc}_r(S, D).e$ is well-defined. As $n > m$, by Definition 2.4.5, we have $\mathrm{Inc}_r(S, D).e = \mathrm{Inc}_r(S, D.s)$

  The dimension of $D.s$ is $n - 1$, hence we get $(n - 1) - m = k - 1$ and by $N(k - 1)$ we obtain that $\mathrm{Inc}_r(S, D.s)$ is well-defined as the inclusion for two well-defined diagrams whose difference in dimensions is $k - 1$.

- For every $0 \leq j < |D|$ we have:

$$\begin{aligned}
&(S \circ D)[j + \mathrm{Inc}_r(S, D).h].g \\
&\quad = (S \circ D)[j + \mathrm{id}_D.h].g && \text{[Def. 2.4.5]} \\
&\quad = (S \circ D)[j].g && \text{[Def. 2.4.1]} \\
&\quad = D[j].g && \text{[Eq. (2.15)]}
\end{aligned}$$

  As required.

- For every $0 \leq j < |D|$ we have:

$$\begin{aligned}
&(\mathrm{Inc}_r(S, D).e).\Lambda[D[j].d] \circ D[j].e = \\
&\quad = \mathrm{Inc}_r(S, D[j].d) \circ D[j].e && [M(k)] \\
&\quad = (S \circ D)[j].e && \text{[Eq. (2.16)]} \\
&\quad = (S \circ D)[j + \mathrm{Inc}_r(S, D).h].e && \text{[Def. 2.4.5]}
\end{aligned}$$

  As required.

As all conditions are satisfied, we can conclude that, hence $\mathrm{Inc}_r(S, D)$ is well-defined.

The argument for $n < m$ that $\mathrm{Inc}_l(S, D)$ is well-defined is analogous. This establishes that $N(k)$ holds and the implication is true. □

Lemma 2.4.17. *For $n \geq 1$ the following holds $K(k) \implies M(k)$*

Proof. We assume that $K(n)$ holds *i.e.* that for any n-diagram D and any m-diagram S such that $|n - m| = k$ and such that the composite $S \circ D$ exists, the following equalities hold:

$$\begin{aligned}
&\text{If } n > m && (S \circ D)[j].d = S \circ (D[j].d) && \text{for any } 0 \leq j < |D| \\
&\text{If } n < m && (S \circ D)[j].d = (S[j].d) \circ D && \text{for any } 0 \leq j < |S|
\end{aligned}$$

We need to show that for any well-defined n-diagram D and a well-defined diagram m-diagram S such that $|n - m| = k$, the following equalities hold:

$$\begin{aligned}
&\text{If } n > m && \mathrm{Inc}_r(S, D[j].d) = (\mathrm{Inc}_r(S, D).e).\Lambda[D[j].d] && \text{for any } 0 \leq j < |D| \\
&\text{If } n < m && \mathrm{Inc}_l(S[j].d, D) = (\mathrm{Inc}_l(S, D).e).\Lambda[S[j].d] && \text{for any } 0 \leq j < |S|
\end{aligned}$$

We consider both cases above separately, first let $n > m$. As $n \geq 1$, to show that these two embeddings are equivalent, by Definition 2.3.6, we need to check three conditions:

- Domains and codomains are equivalent diagrams:

  ⋆ By Definitions 2.3.1, 2.4.5 the type of $\mathrm{Inc}_r(S, D[i].d)$ is as follows:

  $$\mathrm{Inc}_r(S, D[i].d) : D[i].d \hookrightarrow S \circ (D[i].d)$$

  ⋆ By Definitions 2.3.1, 2.4.5 the type of $(\mathrm{Inc}_r(S, D).e).\Lambda[D[i].d]$ is as follows:

  $$\begin{aligned}&(\mathrm{Inc}_r(S, D).e).\Lambda[D[i].d] :\\ &\qquad D[i].d \hookrightarrow ((S \circ D)[\mathrm{Inc}_r(S, D).h]).\Pi[\mathrm{Inc}_r(S, D).e, D[i].d]\end{aligned}$$

  We see that the domains of both are immediately equivalent. For codomains, we need to simplify first. In this derivation we make use of statement $S(n)$ defined in Definition 2.3.10, which holds by Theorem 2.3.27.

  $$\begin{aligned}&((S \circ D)[\mathrm{Inc}_r(S, D).h]).\Pi[\mathrm{Inc}_r(S, D).e, D[i].d] =\\ &\quad = ((S \circ D)[\mathrm{Inc}_r(S, D).h]).\Pi[\mathrm{Inc}_r(S, D).e,\\ &\qquad\quad D[i - \mathrm{Inc}_r(S, D).h].d] && [\text{Def. } 2.4.5]\\ &\quad = (S \circ D).\Pi[(\mathrm{Inc}_r(S, D)), D][i].d && [S(n)]\\ &\quad = (S \circ D)[i].d && [\text{Lemma } 2.3.5]\end{aligned}$$

  Need $S \circ D$ well-defined for this.

  Since $K(n)$ holds, we now obtain that the codomains are equivalent diagrams.

- Component embeddings are equivalent. In this derivation we make use of statement $T(n)$ defined in Definition 2.3.9, which holds by Theorem 2.3.27.

  $$\begin{aligned}&((\mathrm{Inc}_r(S, D).e).\Lambda[D[i].d]).e =\\ &\quad = (\mathrm{Inc}_r(S, D).e).e && [\text{Def. } 2.3.1]\\ &\quad = (\mathrm{Inc}_r(S, D.s)).e && [\text{Def. } 2.4.5]\\ &\quad = \mathrm{Inc}_r(S, D.s.s) && [\text{Def. } 2.4.5]\\ &\quad = \mathrm{Inc}_r(S, (D[i].d).s) && [T(n)]\\ &\quad = (\mathrm{Inc}_r(S, D[i].d)).e && [\text{Def. } 2.4.5]\end{aligned}$$

- Heights are equal:

  $$\begin{aligned}&((\mathrm{Inc}_r(S, D).e).\Lambda[D[i].d]).h =\\ &\quad = (\mathrm{Inc}_r(S, D).e).h && [\text{Def. } 2.3.1]\\ &\quad = (\mathrm{Inc}_r(S, D.s)).h && [\text{Def. } 2.4.5]\\ &\quad = \mathrm{id}_{D.s}.h && [\text{Def. } 2.4.5]\\ &\quad = 0\\ &\quad = \mathrm{id}_{D[i].d}.h && [\text{Def. } 2.4.5]\\ &\quad = (\mathrm{Inc}_r(S, D[i].d)).h && [\text{Def. } 2.4.5]\end{aligned}$$

As all these conditions are fulfilled, the two embeddings are equivalent. The argument for $n < m$ is analogous. This establishes that $M(k)$ holds and the implication is true.

□

Lemma 2.4.18. *For $n \geq 1$ the following holds: $M(k-1) \implies K(k)$*

Proof. Assume that $M(k-1)$ holds, *i.e.* for any well-defined n-diagram D and a well-defined diagram m-diagram S such that $|n-m| = k$ the following equalities hold:

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

We need to show that for any n-diagram D and m-diagram S such that the composite $S \circ D$ exists and $|n-m| = k$ and such that the composite $S \circ D$ exists, the following equalities hold:

If $n > m$	$(S \circ D)[j].d = S \circ (D[j].d)$	for any $0 \leq j < \|D\|$
If $n < m$	$(S \circ D)[j].d = (S[j].d) \circ D$	for any $0 \leq j < \|S\|$

We consider both cases above separately, first let $n > m$. We prove this result by induction on $0 \leq j \leq |D|$.

- *Base case:* For $j = 0$, the result follows immediately from the definitions:

$$\begin{aligned}
&(S \circ D)[0].d = \\
&\quad = (S \circ D).s && \text{[Def. 2.2.5]} \\
&\quad = S \circ (D.s) && \text{[Def. 2.4.6]} \\
&\quad = S \circ (D[0].d) && \text{[Def. 2.2.5]}
\end{aligned}$$

- *Inductive step:* For $j > 0$, assume that:

$$(S \circ D)[j].d = S \circ (D[j].d) \qquad (IH)$$

  Let us consider $(S \circ D)[j+1].d$ and $S \circ (D[j+1].d)$, to show that two diagrams are equivalent, by Definition 2.3.7, we need to check the following:

  ⋆ Sources are equivalent diagrams:

$$\begin{aligned}
&((S \circ D)[j+1].d).s \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, t((S \circ D)[j].g)]).s && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d).s && \text{[Eq. (2.5)]} \\
&\quad = (S \circ (D[j].d)).s && [IH] \\
&\quad = S \circ (D[j].d).s && \text{[Eq. (2.13)]} \\
&\quad = S \circ ((D[j].d).\Pi[D[j].e, t(D[j].g)]).s && \text{[Eq. (2.5)]} \\
&\quad = S \circ (D[j+1].d).s && \text{[Def. 2.2.3]} \\
&\quad = (S \circ (D[j+1].d)).s && \text{[Eq. (2.13)]}
\end{aligned}$$

  ⋆ Lengths of generator lists are equal:

$$\begin{aligned}
&|(S \circ D)[j+1].d| \\
&\quad = |(S \circ D)[j].d).\Pi[(S \circ D)[j].e, t((S \circ D)[j].g)]| && \text{[Def. 2.2.3]} \\
&\quad = |S \circ D)[j].d| - |s((S \circ D)[j].g)| + |t((S \circ D)[j].g)| && \text{[Eq. (2.6)]} \\
&\quad = |S \circ (D[j].d)| - |s((S \circ D)[j].g| + |t((S \circ D)[j].g)| && [IH] \\
&\quad = |S \circ (D[j].d)| - |s(D[j].g| + |t(D[j].g| && \text{[Eq. (2.15)]} \\
&\quad = |D[j].d| - |s(D[j].g| + |t(D[j].g| && \text{[Eq. (2.14)]} \\
&\quad = |D[j+1].d| && \text{[Eq. (2.6)]} \\
&\quad = |S \circ (D[j+1].d)| && \text{[Eq. (2.14)]}
\end{aligned}$$

- For generators and embeddings we need to show that for $0 \leq k < |(S \circ D)[j+1].d| = |S \circ (D[j+1].d)|$, the k-th generators in generator lists of both diagrams correspond and the same for k-th embeddings. Firstly, we distinguish between cases for $m = n-1$ and $m < n-1$.

  Since $m = n-1$, we can simplify the height of the j-th embedding in $S \circ D$ in the following way.

$$\begin{aligned}
&(S \circ D)[j].e.h \\
&\quad = (\mathrm{Inc}_r(S, D[j].d) \circ D[j].e).h && \text{[Eq. (2.12)]} \\
&\quad = (\mathrm{Inc}_r(S, D[j].d)).h + D[j].e.h && \text{[Def. 2.3.2]} \\
&\quad = \mathrm{id}_D.h + D[j].e.h && \text{[Def. 2.4.5]} \\
&\quad = D[j].e.h && \text{[Def. 2.4.1]}
\end{aligned}$$

  Let us refer to this equality as $[*]$.

  We consider these generators and embeddings in three separate ranges:

  ⋆ Range:

$$0 \leq k < (S \circ D)[j].e.h$$

  By $[*]$ this is equivalent to:

$$0 \leq k < D[j].e.h$$

  Generators:

$$\begin{aligned}
&((S \circ D)[j+1].d)[k].g \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\qquad t((S \circ D)[j].g)])[k].g && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d)[k].g && \text{[Eq. (2.7)]} \\
&\quad = (S \circ (D[j].d))[k].g && [IH] \\
&\quad = (D[j].d)[k].g && \text{[Eq. (2.15)]} \\
&\quad = (D[j].d).\Pi[D[j].d, t(D[j].g)])[k].g && \text{[Eq. (2.7)]} \\
&\quad = (D[j+1])[k].g && \text{[Def. 2.2.3]} \\
&\quad = (S \circ (D[j+1].d))[k].g && \text{[Eq. (2.15)]}
\end{aligned}$$

  Embeddings:

$$\begin{aligned}
&((S \circ D)[j+1].d)[k].e \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\qquad t((S \circ D)[j].g)])[k].e && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d)[k].e && \text{[Eq. (2.8)]} \\
&\quad = (S \circ (D[j].d))[k].e && [IH] \\
&\quad = \mathrm{Inc}_r(S, (D[j].d)[k].d) \circ (D[j].d)[k].e && \text{[Eq. (2.16)]} \\
&\quad = \mathrm{Inc}_r(S, (D[j].d.\Pi[D[j].e, t(D[j].g)])[k].d) \circ (D[j].d)[k].e && [S(n-1)] \\
&\quad = \mathrm{Inc}_r(S, (D[j].d.\Pi[D[j].e, t(D[j].g)])[k].d) \\
&\qquad\qquad \circ ((D[j].d.\Pi[D[j].e, t(D[j].g)])[k].d)[k].e && \text{[Eq. (2.8)]} \\
&\quad = \mathrm{Inc}_r(S, (D[j+1].d)[k].d) \circ (D[j+1].d)[k].e && \text{[Def. 2.2.3]} \\
&\quad = (S \circ (D[j+1].d))[k].e && \text{[Eq. (2.16)]}
\end{aligned}$$

⋆ Range:

$$(S \circ D)[j].e.h \leq k < (S \circ D)[j].e.h + |t((S \circ D)[j].g)|$$

By [∗] this is equivalent to

$$D[j].e.h \leq k < D[j].e.h + |t(D[j].g)|$$

Generators:

$$\begin{aligned}
&((S \circ D)[j+1].d)[k].g \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\quad t((S \circ D)[j].g)])[k].g && \text{[Def. 2.2.3]} \\
&\quad = t((S \circ D)[j].g)[k - (S \circ D)[j].e.h].g && \text{[Eq. (2.7)]} \\
&\quad = t(D[j].g)[k - (S \circ D)[j].e.h].g && \text{[Eq. (2.15)]} \\
&\quad = t(D[j].g)[(k - D[j].e.h].g && [*] \\
&\quad = (D[j].d).\Pi[D[j].d, t(D[j].g)])[k].g && \text{[Eq. (2.7)]} \\
&\quad = (D[j+1])[k].g && \text{[Def. 2.2.3]} \\
&\quad = (S \circ (D[j+1].d))[k].g && \text{[Eq. (2.15)]}
\end{aligned}$$

Embeddings: In this derivation we make use of statement $Q(n)$ described in Definition 2.3.11, correctness of which is proved in Theorem 2.3.27.

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

Here we instantiate $Q(n)$ for use in this particular application:

$$\begin{aligned}
f &= \mathrm{Inc}_r(S, D[j].d).e & e &= D[j].e.e \\
A &= (D[j].d)[D[j].e.h].d & T &= t(D[j].g)[k - D[j].e.h].d
\end{aligned}$$

Then, the following equality is given by $[Q(n-1)]$:

$$\begin{aligned}
&((\mathrm{Inc}_r(S, D[j].d).e).\Lambda[(D[j].d)[D[j].e.h].d] \circ D[j].e.e).\Lambda[\\
&\qquad t(D[j].g)[k - D[j].e.h].d] \\
&= (\mathrm{Inc}_r(S, D[j].d).e).\Lambda[((D[j].d)[D[j].e.h].\Pi[D[j].e.e, \\
&\qquad t(D[j].g)[k - D[j].e.h].d])] \circ (D[j].e.e).\Lambda[t(D[j].g)[k - D[j].e.h].d]
\end{aligned}$$

We also make use of the statement $S(n)$ defined in Definition 2.3.10 and which holds by Theorem 2.3.27.

$$\begin{aligned}
&((S \circ D)[j+1].d)[k].e \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, t((S \circ D)[j].g)])[k].e && \text{[Def. 2.2.3]} \\
&\quad = (((S \circ D)[j].e).e).\Lambda[t((S \circ D)[j].g)[k - (S \circ D)[j].e.h].d] \circ \\
&\qquad\quad t((S \circ D)[j].g)[k - (S \circ D)[j].e.h].e && \text{[Eq. (2.8)]} \\
&\quad = (((S \circ D)[j].e).e).\Lambda[t(D[j].g)[k - (S \circ D)[j].e.h].d] \circ \\
&\qquad\quad t(D[j].g)[k - (S \circ D)[j].e.h].e && \text{[Eq. (2.15)]} \\
&\quad = (((S \circ D)[j].e).e).\Lambda[t(D[j].g)[k - D[j].e.h].d] \circ \\
&\qquad\quad t(D[j].g)[k - D[j].e.h].e && [*] \\
&\quad = ((\mathrm{Inc}_r(S, D[j].d) \circ D[j].e).e).\Lambda[t(D[j].g)[k - D[j].e.h].d] \circ \\
&\qquad\quad t(D[j].g)[k - D[j].e.h].e && \text{[Eq. (2.16)]} \\
&\quad = ((\mathrm{Inc}_r(S, D[j].d).e).\Lambda[(D[j].d)[D[j].e.h].d] \circ D[j].e.e).\Lambda[\\
&\qquad\quad t(D[j].g)[k - D[j].e.h].d] \circ t(D[j].g)[k - D[j].e.h].e && \text{[Eq. (2.3.2)]]}
\end{aligned}$$

$$
\begin{aligned}
&= (\mathrm{Inc}_r(S, D[j].d).e).\Lambda[((D[j].d)[D[j].e.h].\Pi[D[j].e.e,\\
&\qquad t(D[j].g)[k - D[j].e.h].d])] \circ (D[j].e.e).\Lambda[t(D[j].g)[k - D[j].e.h].d]\\
&\qquad\qquad \circ\, t(D[j].g)[k - D[j].e.h].e && [Q(n-1)]\\
&= (\mathrm{Inc}_r(S, D[j].d).e).\Lambda[((D[j].d).\Pi[D[j].e,\\
&\qquad t(D[j].g)])[k].d] \circ (D[j].e.e).\Lambda[t(D[j].g)[k - D[j].e.h].d]\\
&\qquad\qquad \circ\, t(D[j].g)[k - D[j].e.h].e && [S(n-1)]\\
&= (\mathrm{Inc}_r(S, (D[j].d).s)).\Lambda[((D[j].d).\Pi[D[j].e,\\
&\qquad t(D[j].g)])[k].d] \circ (D[j].e.e).\Lambda[t(D[j].g)[k - D[j].e.h].d]\\
&\qquad\qquad \circ\, t(D[j].g)[k - D[j].e.h].e && [\text{Def. 2.4.5}]\\
&= (\mathrm{Inc}_r(S, ((D[j].d).\Pi[D[j].e, t(D[j].g)]).s)).\Lambda[((D[j].d).\Pi[D[j].e,\\
&\qquad t(D[j].g)])[k].d] \circ (D[j].e.e).\Lambda[t(D[j].g)[k - D[j].e.h].d]\\
&\qquad\qquad \circ\, t(D[j].g)[k - D[j].e.h].e && [\text{Eq. (2.5)}]\\
&= (\mathrm{Inc}_r(S, (D[j].d).\Pi[D[j].e, t(D[j].g)]).e).\Lambda[((D[j].d).\Pi[D[j].e,\\
&\qquad t(D[j].g)])[k].d] \circ (D[j].e.e).\Lambda[t(D[j].g)[k - D[j].e.h].d]\\
&\qquad\qquad \circ\, t(D[j].g)[k - D[j].e.h].e && [\text{Def. 2.4.5}]\\
&= \mathrm{Inc}_r(S, ((D[j].d).\Pi[D[j].e, t(D[j].g)])[k].d)\\
&\qquad \circ (D[j].e.e).\Lambda[t(D[j].g)[k - D[j].e.h].d] \circ t(D[j].g)[k - D[j].e.h].e && [M(k-1)]\\
&= \mathrm{Inc}_r(S, ((D[j].d).\Pi[D[j].e, t(D[j].g)])[k].d)\\
&\qquad \circ (((D[j].d).\Pi[D[j].e, t(D[j].g)])[k].d)[k].e && [\text{Eq. (2.8)}]\\
&= \mathrm{Inc}_r(S, (D[j+1].d)[k].d) \circ (D[j+1].d)[k].e && [\text{Def. 2.2.3}]\\
&= (S \circ (D[j+1].d))[k].e && [\text{Eq. (2.16)}]
\end{aligned}
$$

⋆ Range:

$$
\begin{aligned}
&(S \circ D)[j].e.h + |t((S \circ D)[j].g)| \le k <\\
&\qquad |(S \circ D)[j].d| - |s((S \circ D)[j].g)| + |t((S \circ D)[j].g)|
\end{aligned}
$$

By [∗] this is equivalent to:

$$
D[j].e.h + |t(D[j].g)| \le k < |D[j].d| - |s(D[j].g)| + |t(D[j].g)|
$$

Generators:

$$
\begin{aligned}
&((S \circ D)[j+1].d)[k].g\\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e,\\
&\qquad\quad t((S \circ D)[j].g)])[k].g && [\text{Def. 2.2.3}]\\
&\quad = ((S \circ D)[j].d)[k - |t((S \circ D)[j].g)| + |s((S \circ D)[j].g)|].g && [\text{Eq. (2.7)}]\\
&\quad = (S \circ (D[j].d))[k - |t((S \circ D)[j].g)| + |s((S \circ D)[j].g)|].g && [IH]\\
&\quad = (S \circ (D[j].d))[(k - |t(D[k].g)| + |s(D[k].g)|)].g && [\text{Eq. (2.15)}]\\
&\quad = (D[j].d)[(k - |t(D[k].g)| + |s(D[k].g)|)].g && [\text{Eq. (2.15)}]\\
&\quad = (D[j].d)[k - |t(D[k].g)| + |s(D[k].g)|].g\\
&\quad = (D[j].d.\Pi[D[j].d, t(D[j].g)])[k].g && [\text{Def. 2.7}]\\
&\quad = (D[j+1].d)[(k - |S|].g && [\text{Def. 2.2.3}]\\
&\quad = (S \circ (D[j+1].d))[k].g && [\text{Eq. (2.15)}]
\end{aligned}
$$

Embeddings:

$$
\begin{aligned}
&((S \circ D)[j+1].d)[k].e \\
&\quad = ((S \circ D)[j].d).\Pi[(S \circ D)[j].e, \\
&\qquad\quad t((S \circ D)[j].g)])[k].e && \text{[Def. 2.2.3]} \\
&\quad = ((S \circ D)[j].d)[k - |t((S \circ D)[j].g)| + |s((S \circ D)[j].g)|].e && \text{[Eq. (2.8)]} \\
&\quad = (S \circ (D[j].d))[k - |t((S \circ D)[j].g)| + |s((S \circ D)[j].g)|].e && [IH] \\
&\quad = (S \circ (D[j].d))[(k - |t(D[k].g)| + |s(D[k].g)|)].e && \text{[Eq. (2.15)]} \\
&\quad = \mathrm{Inc}_r(S, (D[j].d)[k + |s(D[j].g)| - |t(D[j].g)|]) \\
&\qquad\quad \circ (D[j].d)[k - |t(D[k].g)| + |s(D[k].g)|)].e && \text{[Eq. (2.16)]} \\
&\quad = \mathrm{Inc}_r(S, ((D[j].d).\Pi[D[j].e, t(D[j].g)])[k].d) \\
&\qquad\quad \circ (D[j].d)[k - |t(D[k].g)| + |s(D[k].g)|].e && [S(n-1)] \\
&\quad = \mathrm{Inc}_r(S, ((D[j].d).\Pi[D[j].e, t(D[j].g)])[k].d) \\
&\qquad\quad \circ (((D[j].d).\Pi[D[j].e, t(D[j].g)])[k].d)[k].e && \text{[Eq. (2.8)]} \\
&\quad = \mathrm{Inc}_r(S, (D[j+1].d)[k].d) \circ (D[j+1].d)[k].e && \text{[Def. 2.2.3]} \\
&\quad = (S \circ (D[j+1].d))[k].e && \text{[Eq. (2.16)]}
\end{aligned}
$$

All embeddings and generators in both diagrams correspond. Hence, as all these conditions are satisfied, the two diagrams are equivalent by Definition 2.3.7. The argument for $n < m$ is analogous.

By this, we established that $K(k)$ holds, hence the implication is true.

□

Finally, we bring all these lemmas together to prove:

Theorem 2.4.19. *For $k \geq 1$ the following logical statement holds: $K(k) \land L(k) \land M(k) \land N(k)$*

Proof. We prove this by induction on k

- *Base case*: For $k = 1$:
  - ⋆ $K(1)$, holds with no further conditions by Lemma 2.4.14.
  - ⋆ To establish $M(1)$, by Lemma 2.4.17, we need $K(1)$. This holds by the argument above.
  - ⋆ To establish $L(1)$, by Lemma 2.4.15, we need $N(0)$ and L(0) to hold. $L(0)$ holds by the argument above. $N(0)$ holds by Lemma 2.4.12, since $L(0)$ holds.
  - ⋆ To establish $N(1)$, by Lemma 2.4.16, we need all $L(1)$, $M(1)$, $N(0)$ to hold. All hold by the argument above.
- *Inductive step:* For $k > 1$, we assume that all $K(k-1)$, $L(k-1)$, $M(k-1)$, $N(k-1)$ hold.
  - ⋆ To establish $K(k)$, by Lemma 2.4.18, we need $M(k-1)$. This holds by the inductive hypothesis.
  - ⋆ To establish $M(k)$, by Lemma 2.4.17, we need $K(k)$. This holds by the argument above.
  - ⋆ To establish $L(k)$, by Lemma 2.4.15, we need $N(k-1)$ and L(k-1) to hold. Both hold by the inductive hypothesis.
  - ⋆ To establish $N(k)$, by Lemma 2.4.16, we need all $L(k)$, $M(k)$, $N(k-1)$ to hold. The initial two statements hold by the argument above. $N(k-1)$ holds by the inductive hypothesis.

  As all statements $K(k)$, $L(k)$, $M(k)$, $N(k)$ hold, this establishes that their conjunction is true.

By this inductive argument the logical statement: $K(k) \land L(k) \land M(k) \land N(k)$ holds for $k \geq 1$.

□

The two main results on the composite of two diagrams being well-defined and the inclusion embedding being well-defined follow immediately:

Theorem 2.4.20. *For any well-defined n-diagram D and a well-defined diagram m-diagram S such that $|n-m| = k$:*

- *If $n \geq m$ and $t(S) = s^{n-m+1}(D)$ and the composite $S \circ D$ exists, the inclusion embedding $Inc_r(S, D) : D \hookrightarrow S \circ D$ is well-defined.*
- *If $n < m$ and $t^{m-n+1}(S) \equiv s(D)$ and the composite $S \circ D$ exists, the inclusion embedding $Inc_r(S, D) : D \hookrightarrow D \circ S$ is well-defined.*

Proof. By Lemma 2.4.12 this holds for $n = m$ and by Theorem 2.4.19 for $n \neq m$. □

Theorem 2.4.21. *For any well-defined n-diagram D and a well-defined m-diagram S such that $|n-m| = k$ and $t(S) = s^{n-m+1}(D)$ if $m \leq n$ or $t^{m-n+1}(S) = s(D)$ otherwise, then the* composite *diagram $S \circ D$ is well-defined*

Proof. By Lemma 2.4.13 this holds for $n = m$ and by Theorem 2.4.19 for $n \neq m$. □

2.4.4 Identity diagrams

There are instances where it is desirable to reason about a composed sequence of rewrites of an n-diagram D having no overall effect on D, because of that we define the notion of an *identity* operation on D. We can use it to talk about equivalence between the identity on D and a sequence of $(n+1)$-rewrites, *i.e.* the sequence having no effect on D.

Definition 2.4.22. Given an n-diagram D the *identity diagram* $\mathrm{Id}(D)$ on D, is the following $(n+1)$-diagram:

$$\mathrm{Id}(D).s = D$$
$$|\mathrm{Id}(D)| = 0$$

Lemma 2.4.23. *Given a well-defined n-diagram D the identity* $\mathrm{Id}(D)$ *on the diagram D is well-defined.*

Proof. Since $\mathrm{Id}(D).s = \mathrm{Id}(D)[0].d$ it is the only and final slice, there is no signature element and no embedding associated with it. $\mathrm{Id}(D).s = D$ is well-defined as D is well-defined, hence by Definition 2.2.6 $\mathrm{Id}(D)$ is also well-defined. □

We also refer to this operation as *boosting* a diagram D. As expected, composing an n-diagram D with an identity on an m-diagram S such that $n > m$ leaves D unaltered. This is because of the requirements on matches between sources and targets of the diagrams being composed.

Lemma 2.4.24. *Given a well-defined n-diagram D and a well-defined m-diagram S such that $m < n$, the following holds:*

- *If $S = \mathrm{Id}(S).t = s^{n-m+1}(D)$:*

$$\mathrm{Id}(S) \circ D = D$$

- *If $t^{n-m+1}(D) = \mathrm{Id}(S).s = S$:*

$$D \circ \mathrm{Id}(S) = D$$

Proof. First let us assume that $S = s^{n-m+1}(D)$. We show that $\mathrm{Id}(S) \circ D$ and D are equivalent diagrams by induction on $k = (n-1) - m$.

- *Base case*: For $k = 0$, we have $n - 1 = m$. By Definition 2.3.7, we need to show four separate conditions:

  - ⋆ Sources are equivalent diagrams:

$$
\begin{aligned}
&(\mathrm{Id}(S) \circ D).s \\
&\quad = \mathrm{Id}(S).s && \text{[Eq. (2.9)]} \\
&\quad = S && \text{[Def. 2.4.22]} \\
&\quad = \mathrm{Id}(S).t && \text{[Def. 2.4.22]} \\
&\quad = D.s && \text{[assumption]}
\end{aligned}
$$

  - ⋆ Sizes of generator lists are equal:

$$
\begin{aligned}
&|\mathrm{Id}(S) \circ D| \\
&\quad = |\mathrm{Id}(S)| + |D| && \text{[Eq. (2.10)]} \\
&\quad = |D| && \text{[Def. 2.4.22]}
\end{aligned}
$$

  - ⋆ Corresponding generators are equal, for $0 \leq j \leq |D|$:

$$
\begin{aligned}
&(\mathrm{Id}(S) \circ D)[j].g \\
&\quad = D[j].g && [\text{Eq. (2.11)}, |\mathrm{Id}(S)| = 0]
\end{aligned}
$$

  - ⋆ Corresponding embeddings are equivalent, for $0 \leq j \leq |D|$:

$$
\begin{aligned}
&(\mathrm{Id}(S) \circ D)[j].e \\
&\quad = D[j].e && [\text{Eq. (2.12)}, |\mathrm{Id}(S)| = 0]
\end{aligned}
$$

- *Inductive step*: For $k > 0$, we assume that the result holds, *i.e.* for all x-diagrams M and y-diagrams M such that $k = (x - 1) - y$, we have $\mathrm{Id}(N) \circ M = M$ (IH).

  Now consider an n-diagram D and an m-diagram S, such that $k + 1 = (n - 1) - m$, then for $\mathrm{Id}(S) \circ D = D$ to hold, by Definition 2.3.7, we need to show four separate conditions:

  - ⋆ Sources are equivalent diagrams:

$$
\begin{aligned}
&(\mathrm{Id}(S) \circ D).s \\
&\quad = \mathrm{Id}(S) \circ D.s && \text{[Eq. (2.13)]} \\
&\quad = D.s && [IH]
\end{aligned}
$$

  - ⋆ Sizes of generator lists are equal:

$$
\begin{aligned}
&|\mathrm{Id}(S) \circ D| \\
&\quad = |D| && \text{[Eq. (2.14)]}
\end{aligned}
$$

  - ⋆ Corresponding generators are equal, for $0 \leq j \leq |D|$:

$$
\begin{aligned}
&(\mathrm{Id}(S) \circ D)[j].g \\
&\quad = D[j].g && \text{[Eq. (2.15)]}
\end{aligned}
$$

  - ⋆ Corresponding embeddings are equivalent, for $0 \leq j \leq |D|$:

$$
\begin{aligned}
&(\mathrm{Id}(S) \circ D)[j].e \\
&\quad = \mathrm{Inc}_r(\mathrm{Id}(S), D[j].d) \circ D[j].e && \text{[Eq. (2.16)]} \\
&\quad = D[j].e && [IH]
\end{aligned}
$$

By this inductive argument, we established that for any n-diagram D and any m-diagram S such that $m < n$, such that $S = s^{n-m+1}(D)$ the following holds: $\mathrm{Id}(S) \circ D = D$.

The argument for $S = t^{n-m+1}((D)$ is analogous, so the entire result holds. □

However, if the diagram S that the identity operation acts on is of the same dimension n as D, we get slightly different behaviour.

Lemma 2.4.25. *Given a well-defined n-diagram D and a well-defined diagram S such that $m, n > 0$, the following holds:*

- *If $S = s^{n-m+1}(D)$:*

$$S \circ \mathrm{Id}(D) = \mathrm{Id}(S \circ D)$$

- *If $t^{m-n+1}(S) = D$:*

$$\mathrm{Id}(S) \circ D = \mathrm{Id}(S \circ D)$$

Proof. We prove the result for both cases separately. If $S = s^{n-m+1}(D)$, we have $n \geq m$.

Since $n, m > 0$ to show that these two diagrams are equivalent, by Definition 2.3.7, we need to check four separate conditions:

- Sources are quivalent diagrams:

$$\begin{aligned}
&(S \circ \mathrm{Id}(D) = \mathrm{Id}(S \circ D)).s \\
&\quad = S \circ \mathrm{Id}(D).s && \text{[Eq. (2.17)]} \\
&\quad = S \circ D && \text{[Def. 2.4.22]} \\
&\quad = \mathrm{Id}(S \circ D).s && \text{[Def. 2.4.22]}
\end{aligned}$$

- Sizes of generator lists are equal:

$$\begin{aligned}
&|\mathrm{Id}(S) \circ D| \\
&\quad = |\mathrm{Id}(S)| && \text{[Eq. (2.18)]} \\
&\quad = 0 && \text{[Def. 2.4.22]} \\
&\quad = |\mathrm{Id}(S \circ D)| && \text{[Def. 2.4.22]}
\end{aligned}$$

- Since $|\mathrm{Id}(S) \circ D| = 0$, we do not need to show anything further for generators and embeddings and the remaining two conditions are vacuously true.

This establishes that the two diagrams are equivalent, as required. The argument for $t^{n-m+1}(D) = S$ is analogous.

□

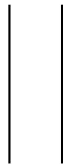
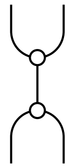
2.5 Associativity and distributivity of diagram composition

With the goal of modelling quasistrict n-categories in mind and taking into account that the only non-trivial morphisms that we want to keep are the interchange law and coherences derived from it, we do not want to include associator morphisms. For this reason, we need to show that certain associativity and distributivity results are built-in properties of diagram composition.

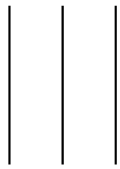
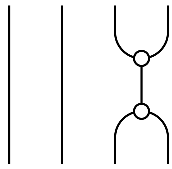
Let us consider different possible ways in which three diagrams can be composed. For any three well-defined diagrams: an n-diagram D, m-diagram S and an l-diagram M the form of the composite depends on the order of binary compositions (bracketing) and the ordering of natural numbers m, n, l.

Certain combinations allow for associativity or distributivity rules, others do not yield any interesting behaviour. Before we proceed to listing these formally, we give several examples to illustrate how associativity or distributivity of diagram composition arises.

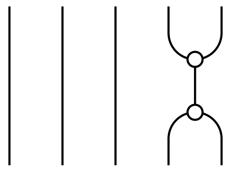
Firstly, consider the following 1-diagrams S, D, and a 2-diagram M, note that S and D are of the same dimension:

$S \quad =$ $\quad D \quad =$  $\quad M \quad =$ 

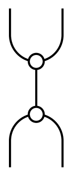
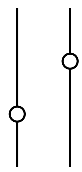
We could then form the following composites:

$S \circ D \quad =$  $\quad D \circ M \quad =$ 

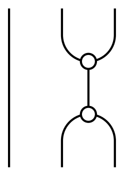
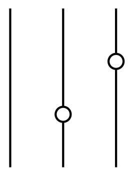
Now note that the order (bracketing) in which we decided to perform the binary compositions does not have any effect on the final result:

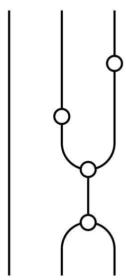
$S \circ (D \circ M) = (S \circ D) \circ M =$ 

Secondly, let D and M have the same dimensions which are different than the dimension of S:

$S \quad =$ $\quad D \quad =$  $\quad M \quad =$ 

This results in different behaviour, as composition with S distributes over composition of diagrams D and M, *i.e.* we could first separately compose S with D and with M and then compose the resulting diagrams vertically, or alternatively we could first vertically compose D with M and then compose the result with S:

$S \circ D \quad =$  $\quad S \circ M \quad =$ 

$S \circ (D \circ M) = (S \circ D) \circ (S \circ M) \quad =$ 

At this point let us remind ourselves that each composite can either be denoted explicitly by the dimensions of the diagrams involved, such as $S \circ_{m,n} D$ or using an overloaded notation $S \circ_{\min(m,n)-1} D$. Below we present a theorem that summarises all the interesting associativity and distributivity laws for composition of three diagrams. The most interesting clauses are proved in the later part of this section, through the familiar technique of making several logical statements depending on a natural number k (in this instance, the difference between diagram dimensions) and then proving a conjunction of all these statements by induction on k.

Theorem 2.5.1. *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:*

$$S \circ_a (D \circ_a M) = (S \circ_a D) \circ_a M \qquad \text{if } a = b \tag{2.21}$$

$$S \circ_b (D \circ_a M) = (S \circ_b D) \circ_a (S \circ_b M) \qquad \text{if } b < a \tag{2.22}$$

$$(S \circ_c D) \circ_d M = (S \circ_d M) \circ_c (D \circ_d M) \qquad \text{if } d < c \tag{2.23}$$

Proof. We consider these three equalities separately:

- For equality Eq. (2.21), we have $a = b$, this implies $\min(n,l) = \min(m, \max(n,l))$, which in turn forces one of the following three options:

  ⋆ $n = m \leq l$

  ⋆ $n = l \leq m$

  ⋆ $m = l \leq n$

  We prove the equality for the first of these cases in Lemma 2.5.7. The setup is prepared by Definitions 2.5.2 and 2.5.3. The remaining two cases follow by an analogous argument.

- For equality Eq. (2.22), we have $a > b$, this implies $\min(n,l) > \min(m, \max(n,l))$, which in turn forces $n, l > m$. We prove this in Lemma 2.5.13, after preparing the setup by Definitions 2.5.8, A.0.16.

- For equality Eq. (2.23), similarly we have $c > d$, this implies $\min(n,m) > \min(l, \max(n,m))$, which in turn forces $n, m > l$. The argument for this case is analogous to the proof for equality Eq. (2.22).

□

Instances for $b > a$ or $d > c$ are not included, as they do not give rise to any associativity or distributivity laws. As an example, consider the composite $S \circ_b (D \circ_a M)$ such that $m > n > l$, then $a = l - 1$, $b = n - 1$ and we have $b > a$. Let us consider when such a composite exists. We need the following:

- $t^{n-l+1}(D) = s(M)$
- $t^{m-n+1}(S) = s(D \circ_a M)$

Since $n > l$, by equation 2.13, we obtain that: $s(D \circ M) = D.s \circ M$. By this, we see that the composite of the source of D with M must match the appropriate target boundary of S, so we cannot compose them in any other order, as the relevant sources and targets would not match.

We now prove the results stated in Theorem 2.5.1. First, we make two logical statements which, when established for all $k \geq 0$, prove that equality Eq. (2.21) holds. Apart from the main result on associativity of composition, we additionally need a statement on composition of inclusion embeddings.

Definition 2.5.2. $[E(k)]$ For $k \geq 0$, let $E(k)$ denote the statement that for two well-defined n-diagrams D, S, and a well-defined l-diagram M such that $l > n > 0$ and $l - n = k$, the following holds:

$$S \circ (D \circ M) = (S \circ D) \circ M$$

Definition 2.5.3. [$F(k)$] For $k, n \geq 0$, let $F(k)$ denote the statement that for two well-defined n-diagrams D, S, and a well-defined l-diagram M such that $l > n$ and $l - n = k$, the following holds:

$$\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M) = \mathrm{Inc}_r(S \circ D, M)$$

First, in a separate lemma, we establish the base for the recursive proof for the Lemma 2.5.7, which comes later in the section.

Lemma 2.5.4. *The statement $E(0)$ holds with no further assumptions*

Proof. We need to show that given three well-defined diagrams: an n-diagram D, and m-diagram S and an l-diagram M, the following holds:

$$S \circ (D \circ M) = (S \circ D) \circ M$$

As $k = 0$, this gives us $n = l$. Since $n > 1$, by Definition 2.3.7, we need to check the following four conditions:

- Sources are equivalent diagrams, the derivation follows by Eq. (2.9):

$$\begin{aligned}&(S \circ (D \circ M)).s\\ &\quad = S.s\\ &\quad = (S \circ D).s\\ &\quad = ((S \circ D) \circ M).s\end{aligned}$$

- Sizes of generator lists are equal, the derivation follows by Eq. (2.10):

$$\begin{aligned}&|S \circ (D \circ M)|\\ &\quad = |S| + |D \circ M|\\ &\quad = |S| + |D| + |M|\\ &\quad = |(S \circ D)| + |M|\\ &\quad = |(S \circ D) \circ M|\end{aligned}$$

- Generators are equal for $0 \leq j < |S| + |D| + |M|$, we show this for three separate ranges. First let $0 \leq j < |S|$, the derivation follows by Eq. (2.11):

$$\begin{aligned}&(S \circ (D \circ M))[j].g\\ &\quad = S[j].g\\ &\quad = (S \circ D)[j].g\\ &\quad = ((S \circ D) \circ M)[j].g\end{aligned}$$

  The argument is analogous for the remaining two ranges $|S| \leq j < |S| + |D|$ and $|S| + |D| \leq j < |S| + |D| + |M|$.

- Embeddings are equivalent for $0 \leq j < |S| + |D| + |M|$, we show this for three separate ranges. First let us show this for $|S| \leq j < |S| + |D|$, the derivation follows by Eq. (2.12):

$$\begin{aligned}&(S \circ (D \circ M))[j].e\\ &\quad = (D \circ M)[j - |S|].e\\ &\quad = D[j - |S|].e\\ &\quad = ((S \circ D)[j].e\\ &\quad = ((S \circ D) \circ M)[j].e\end{aligned}$$

  The argument is analogous for the remaining two ranges $0 \leq j < |S|$ and $|S| + |D| \leq j < |S| + |D| + |M|$.

By this argument, the diagrams $S \circ (D \circ M)$ and $(S \circ D) \circ M$ are equivalent and the statement $E(0)$ holds, as required. ☐

The following two implications establish $E(k)$ and $F(k)$:

Lemma 2.5.5. *For $k \geq 1$ the following statement holds: $F(k-1) \wedge E(k-1) \implies E(k)$*

Proof. Let us assume that both $F(k-1)$ and $E(k-1)$ hold. We need to show that given three well-defined diagrams: an n-diagram D, and m-diagram S and an l-diagram M, the following holds:

$$S \circ (D \circ M) = (S \circ D) \circ M$$

We need to show this result for all the possible orderings of n, m, l. First, assume $m < n < l$:
Since $n > 1$, by Definition 2.3.7, we need to check the following four conditions:

- Sources are equivalent diagrams, the derivation follows by Eq. (2.13):

$$\begin{aligned}
&(S \circ (D \circ M)).s \\
&\quad = S \circ (D \circ M).s \\
&\quad = S \circ (D \circ M.s) \\
&\quad = (S \circ D) \circ M.s) && [E(k-1)] \\
&\quad = ((S \circ D) \circ M).s
\end{aligned}$$

- Sizes of generator lists are equal, the derivation follows by Eq. (2.14):

$$\begin{aligned}
&|S \circ (D \circ M)| \\
&\quad = |D \circ M| \\
&\quad = |M| \\
&\quad = |(S \circ D) \circ M|
\end{aligned}$$

- Generators are equal for $0 \leq j \leq |S \circ (D \circ M)|$, the derivation follows by Eq. (2.15):

$$\begin{aligned}
&(S \circ (D \circ M))[j].g \\
&\quad = (D \circ M)[j].g \\
&\quad = M[j].g \\
&\quad = ((S \circ D) \circ M)[j].g
\end{aligned}$$

- Embeddings are equivalent for $0 \leq j \leq |S \circ (D \circ M)|$, the derivation follows by Eq. (2.16):

$$\begin{aligned}
&(S \circ (D \circ M))[j].e \\
&\quad = \mathrm{Inc}_r(S, (D \circ M)[j].d) \circ (D \circ M)[j].e \\
&\quad = \mathrm{Inc}_r(S, (D \circ M)[j].d) \circ (\mathrm{Inc}_r(D, M[j].d) \circ M[j].e) \\
&\quad = (\mathrm{Inc}_r(S, (D \circ M)[j].d) \circ \mathrm{Inc}_r(D, M[j].d)) \circ M[j].e && [A(n), \text{Def. } 2.3.27] \\
&\quad = \mathrm{Inc}_r(S \circ D, M[j].d) \circ M[j].e && [F(n-1)] \\
&\quad = ((S \circ D) \circ M)[j].e
\end{aligned}$$

  We already showed that the statement $A(n)$ holds for all $n \geq 1$ in the proof of the Theorem 2.3.27, so there is no need to include it separately in the conjunction of logical statements being proved here.

As all these conditions are satisfied, we can conclude that both diagrams are equivalent, as required. Arguments for other orderings of n, m and l are analogous. By this, we established that $E(k)$ holds, hence the implication is true.

☐

Lemma 2.5.6. *For $k \geq 1$ the following statement holds: $F(k-1) \wedge E(k) \implies F(k)$, additionally F(0) holds with no further assumptions.*

Proof. Let us assume that both $F(k-1)$ and $E(k)$ hold. We need to show that given two well-defined n-diagrams D, S, and a well-defined l-diagram M such that $l > n > 0$ and $l - n = k$, the following holds:

$$\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M) = \mathrm{Inc}_r(S \circ D, M)$$

Since $l > n > 0$ by Lemma 2.3.6, we need to check the following conditions:

- Types are the same. By Definition 2.4.5, these types are as follows:

$$\begin{aligned} \mathrm{Inc}_r(D, M) &: M \hookrightarrow D \circ M \\ \mathrm{Inc}_r(S, (D \circ M)) &: D \circ M \hookrightarrow S \circ (D \circ M) \\ \mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M) &: M \hookrightarrow S \circ (D \circ M) \\ \mathrm{Inc}_r(S \circ D, M) &: M \hookrightarrow (S \circ D) \circ M \end{aligned}$$

  The domains are immediately equivalent, codomains are equivalent by $E(n)$.

- Heights are equal. For this condition, we consider two scenarios:

  ⋆ For $k = 0$:

$$\begin{aligned} &(\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M)).h \\ &\quad = \mathrm{Inc}_r(S, D \circ M).h + \mathrm{Inc}_r(D, M).h && \text{[Def. 2.3.2]} \\ &\quad = |S| + |D| && \text{[Def. 2.4.5]} \\ &\quad = |S \circ D| && \text{[Eq. (2.10)]} \\ &\quad = (\mathrm{Inc}_r(S \circ D, M)).h && \text{[Def. 2.4.5]} \end{aligned}$$

  ⋆ For $k > 0$:

$$\begin{aligned} &(\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M)).h \\ &\quad = \mathrm{Inc}_r(S, D \circ M).h + \mathrm{Inc}_r(D, M).h && \text{[Def. 2.3.2]} \\ &\quad = \mathrm{id}_{D \circ M}.h + \mathrm{id}_M.h && \text{[Eq. (2.4.5)]} \\ &\quad = 0 \\ &\quad = \mathrm{id}_M.h && \text{[Def. 2.4.5]} \\ &\quad = (\mathrm{Inc}_r(S \circ D, M)).h && \text{[Def. 2.4.5]} \end{aligned}$$

- Component embeddings are equivalent. Here, again we consider two scenarios:

  ⋆ For $k = 0$:

$$\begin{aligned} &(\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M)).e \\ &\quad = (\mathrm{Inc}_r(S, D \circ M).e).\Lambda[(D \circ M)[\mathrm{Inc}_r(D, M).h].d] \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.3.2]} \\ &\quad = (\mathrm{Inc}_r(S, D \circ M).e).\Lambda[(D \circ M)[\mathrm{Inc}_r(D, M).h].d] \circ \mathrm{id}_M.e && \text{[Def. 2.4.5]} \\ &\quad = (\mathrm{id}_{D \circ M}.e).\Lambda[(D \circ M)[\mathrm{Inc}_r(D, M).h].d] \circ \mathrm{id}_M.e && \text{[Def. 2.4.5]} \\ &\quad = \mathrm{id}_{(D \circ M)[\mathrm{Inc}_r(D,M).h].d} \circ \mathrm{id}_M.e && \text{[Def. 2.4.2]} \\ &\quad = \mathrm{id}_{(D \circ M)[|D|].d} \circ \mathrm{id}_M.e && \text{[Def. 2.4.5]} \\ &\quad = \mathrm{id}_{M.s} \circ \mathrm{id}_M.e && \text{[Def. 2.3.2]} \\ &\quad = \mathrm{id}_{M.s} \circ \mathrm{id}_{M.s} && \text{[Def. 2.4.1]} \\ &\quad = \mathrm{id}_{M.s} && \text{[Def. 2.4.3]} \\ &\quad = \mathrm{id}_M.e && \text{[Def. 2.4.1]} \\ &\quad = (\mathrm{Inc}_r(S \circ D, M)).e && \text{[Def. 2.4.5]} \end{aligned}$$

⋆ For $k > 0$:

$$
\begin{aligned}
&(\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M)).e \\
&\quad = (\mathrm{Inc}_r(S, D \circ M).e).\Lambda[(D \circ M)[\mathrm{Inc}_r(D, M).h].d] \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.3.2]} \\
&\quad = (\mathrm{Inc}_r(S, D \circ M).e).\Lambda[(D \circ M)[0].d] \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.4.5]} \\
&\quad = \mathrm{Inc}_r(S, (D \circ M)[0].d) \circ \mathrm{Inc}_r(D, M).e && [M(n)] \\
&\quad = \mathrm{Inc}_r(S, (D \circ M).s) \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.3.2]} \\
&\quad = \mathrm{Inc}_r(S, (D \circ M).s) \circ \mathrm{Inc}_r(D, M.s) && \text{[Def. 2.3.2]} \\
&\quad = \mathrm{Inc}_r(S, D \circ M.s) \circ \mathrm{Inc}_r(D, M.s) && \text{[Eq. (2.13)]} \\
&\quad = \mathrm{Inc}_r(S \circ D, M.s) && [F(n-1)] \\
&\quad = (\mathrm{Inc}_r(S \circ D, M)).e && \text{[Def. 2.4.5]}
\end{aligned}
$$

As with the proof of Lemma 2.5.5, we already proved that the statement $M(n)$ holds for all $n \geq 1$ by Theorem 2.3.27, so there is no need to include it separately in the conjunction of logical statements being proved here.

As all these conditions are satisfied both embeddings are equivalent. By this, we established that $F(k)$ holds, hence the implication is true. Additionally, since we used the assumptions $E(k)$ and $F(k-1)$ only for $n > 0$, F(0) holds with no further assumptions.

□

These three lemmas, allow us to prove that the conjunction of statements E and F holds for all $k \geq 0$, therefore proving that equality Eq. (2.21) holds.

Lemma 2.5.7. *For $k \geq 0$: the following holds: $E(k) \wedge F(k)$*

Proof. We prove this by induction on k:

- *Base case*: For $k = 0$
  - ⋆ $F(0)$ holds by Lemma 2.5.6.
  - ⋆ $E(0)$ holds by LemmA 2.5.4.
- *Inductive step*: For $k > 0$ we assume that both $F(k-1)$ and $E(k-1)$ hold.
  - ⋆ To establish $E(k)$, by Lemma 2.5.5 we need both $E(k-1)$ and $F(k-1)$ to hold. Both hold by the inductive hypothesis.
  - ⋆ To establish $F(k)$, by Lemma 2.5.6 we need both $E(k)$ and $F(k-1)$ to hold. Both hold by the statement above.

By this inductive argument the statement $E(k) \wedge F(k)$ holds for $k \geq 0$.

□

Below, in a similar way to Definitions 2.5.2, 2.5.3, we make two logical statements such that when their conjunction is shown for all $k \geq 0$, equality Eq. (2.22) is proved. Here k is the difference between dimensions of the two diagrams in the bracket.

Definition 2.5.8. $[G(k)]$ For $k \geq 0$, let $G(k)$ denote the statement that for three well-defined diagrams: an n-diagrams D, an m-diagram S and an l-diagram M, such that $l, n > m > 0$ and $|l - n| = k$, the following holds:

$$S \circ_b (D \circ_a M) = (S \circ_b D) \circ_a (S \circ_b M) \qquad \text{if } b < a$$

Here, we have $a = \min(n, l) - 1$, $b = \min(m, \max(n, l)) - 1$.

Definition 2.5.9. $[H(k)]$ For $k \geq 0$,let $H(k)$ denote the statement that for three well-defined diagrams: an n-diagrams D, an m-diagram S and an l-diagram M, such that $l, n > m > 0$ and $|l - n| = k$, then, provided that these composites exist, the following holds:

$$\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M) = \mathrm{Inc}_r(S \circ D, S \circ M) \circ \mathrm{Inc}_r(S, M)$$

Here, we have $a = \min(n, l)$, $b = \min(m, \max(n, l)) - 1$.

Again, in a similar fashion to Lemma 2.5.4, we first establish the statement G for $k = 0$.

Lemma 2.5.10. *The statement $G(0)$ holds without any further assumptions.*

Proof. For $G(0)$ to hold, we need to show that for any three well-defined diagrams: an n-diagrams D, an m-diagram S and an l-diagram M, such that $l, n > m > 0$ and $|l - n| = k$, the following holds:

$$S \circ_b (D \circ_a M) = (S \circ_b D) \circ_a (S \circ_b M)$$

Since $k = |l - n| = 0$, we have $l = n > m$. Bearing that in mind we drop the composition indices. As $m > 0$, by Definition 2.3.7, to show equivalence of these two diagrams, we need to check the following four conditions:

- Sources are equivalent diagrams:

$$\begin{aligned}
&(S \circ (D \circ M)).s \\
&\quad = S \circ (D \circ M).s && \text{[Eq. (2.13)]} \\
&\quad = S \circ D.s && \text{[Eq. (2.9)]} \\
&\quad = (S \circ D).s && \text{[Eq. (2.13)]} \\
&\quad = ((S \circ D) \circ (S \circ M)).s && \text{[Eq. (2.9)]}
\end{aligned}$$

- Sizes of generator and embedding lists for both diagrams are equal:

$$\begin{aligned}
&|S \circ (D \circ M)| \\
&\quad = |D \circ M| && \text{[Eq. (2.14)]} \\
&\quad = |D| + |M| && \text{[Eq. (2.9)]} \\
&\quad = |(S \circ D)| + |(S \circ M)| && \text{[Eq. (2.14)]} \\
&\quad = |(S \circ D) \circ (S \circ M)| && \text{[Eq. (2.9)]}
\end{aligned}$$

- Corresponding generators are equal for $0 \leq j \leq |M| + |D|$, we show this for two separate ranges, first assume $|D| \leq j < |D| + |M|$:

$$\begin{aligned}
&(S \circ (D \circ M))[j].g \\
&\quad = (D \circ M)[j].g && \text{[Eq. (2.15)]} \\
&\quad = M[j - |D|].g && \text{[Eq. (2.11)]} \\
&\quad = (S \circ M)[j - |D|].g && \text{[Eq. (2.15)]} \\
&\quad = (S \circ M)[j - |S \circ D|].g && \text{[Eq. (2.14)]} \\
&\quad = ((S \circ D) \circ (S \circ M))[j].g && \text{[Eq. (2.11)]}
\end{aligned}$$

  The argument for $0 \leq j < |D|$ is analogous.

- Corresponding embeddings are equivalent for $0 \leq j \leq |M| + |D|$, we show this for two separate ranges, first assume $|D| \leq j < |D| + |M|$:

$$
\begin{aligned}
&(S \circ (D \circ M))[j].e \\
&\quad = \mathrm{Inc}_r(S, (D \circ M)[j].d) \circ (D \circ M)[j].e && \text{[Eq. (2.16)]} \\
&\quad = \mathrm{Inc}_r(S, (D \circ M)[j].d) \circ M[j - |D|].e && \text{[Eq. (2.12)]} \\
&\quad = \mathrm{Inc}_r(S, M[j - |D|.d]) \circ M[j - |D|].e && \text{[Def. 2.4.7]} \\
&\quad = (S \circ M)[j - |D|].e && \text{[Eq. (2.10)]} \\
&\quad = (S \circ M)[j - |S \circ D|].e && \text{[Eq. (2.14)]} \\
&\quad = ((S \circ D) \circ (S \circ M))[j].e && \text{[Eq. (2.12)]}
\end{aligned}
$$

  The argument for $0 \leq j < |D|$ is analogous.

Since all these conditions are satisfied, we established that $S \circ (D \circ M) = (S \circ D) \circ (S \circ M)$ for an m-diagram S and an l-diagram M, such that $l = n > m > 0$. Hence, the statement $G(0)$ holds, as required.

□

This is followed by implications establishing $G(k)$ and $H(k)$.

Lemma 2.5.11. *For $k \geq 1$ the following statement holds: $G(k-1) \wedge H(k-1) \implies G(k)$*

Proof. We assume that both $G(k-1)$ and $H(k-1)$ hold.

For $G(k)$ to hold, we need to show that for any three well-defined diagrams: an n-diagrams D, an m-diagram S and an l-diagram M, such that $l, n > m > 0$ and $|l - n| = k$, the following holds:

$$S \circ_b (D \circ_a M) = (S \circ_b D) \circ_a (S \circ_b M)$$

Since $k = |l - n| > 0$, we have $l > n > m$ or $n > l > m$. First, assume $l > n > m$, then bearing that in mind we drop the composition indices. As $m > 0$, by Definition 2.3.7, to show equivalence of these two diagrams, we need to check the following four conditions:

- Sources are equivalent diagrams:

$$
\begin{aligned}
&(S \circ (D \circ M)).s \\
&\quad = S \circ (D \circ M).s && \text{[Eq. (2.13)]} \\
&\quad = S \circ (D \circ M.s) && \text{[Eq. (2.13)]} \\
&\quad = (S \circ D) \circ (S \circ M.s) && [G(k-1)] \\
&\quad = (S \circ D) \circ (S \circ M).s && \text{[Eq. (2.13)]} \\
&\quad = ((S \circ D) \circ (S \circ M)).s && \text{[Eq. (2.13)]}
\end{aligned}
$$

- Sizes of generator and embedding lists for both diagrams are equal:

$$
\begin{aligned}
&|S \circ (D \circ M)| \\
&\quad = |D \circ M| && \text{[Eq. (2.14)]} \\
&\quad = |M| && \text{[Eq. (2.14)]} \\
&\quad = |(S \circ M)| && \text{[Eq. (2.14)]} \\
&\quad = |(S \circ D) \circ (S \circ M)| && \text{[Eq. (2.14)]}
\end{aligned}
$$

- Corresponding generators are equal for $0 \leq j \leq |M|$:

$$\begin{aligned}
&(S \circ (D \circ M))[j].g \\
&\quad = M[j].g && \text{[Eq. (2.15)]} \\
&\quad = (S \circ M)[j].g && \text{[Eq. (2.15)]} \\
&\quad = ((S \circ D) \circ (S \circ M))[j].g && \text{[Eq. (2.15)]}
\end{aligned}$$

- Corresponding embeddings are equal for $0 \leq j \leq |M|$:

$$\begin{aligned}
&(S \circ (D \circ M))[j].e \\
&\quad = \mathrm{Inc}_r(S, (D \circ M)[j].d) \circ (D \circ M)[j].e && \text{[Eq. (2.16)]} \\
&\quad = \mathrm{Inc}_r(S, (D \circ M)[j].d) \circ (\mathrm{Inc}_r(D, M[j].d) \circ M[j].e) && \text{[Eq. (2.16)]} \\
&\quad = (\mathrm{Inc}_r(S, (D \circ M)[j].d) \circ \mathrm{Inc}_r(D, M[j].d)) \circ M[j].e && [A(n)] \\
&\quad = (\mathrm{Inc}_r(S \circ D, (S \circ M)[j].d) \circ \mathrm{Inc}_r(S, M[j].d)) \circ M[j].e && [H(k-1)] \\
&\quad = \mathrm{Inc}_r(S \circ D, (S \circ M)[j].d) \circ (\mathrm{Inc}_r(S, M[j].d) \circ M[j].e) && [A(n)] \\
&\quad = \mathrm{Inc}_r(S \circ D, (S \circ M)[j].d) \circ (S \circ M)[j].e && \text{[Eq. (2.16)]} \\
&\quad = ((S \circ D) \circ (S \circ M))[j].e && \text{[Eq. (2.16)]}
\end{aligned}$$

Here $A(n)$ is the statement on associativity of embedding composition proved in Theorem 2.3.27. Since all these conditions are satisfied, we established that $S \circ (D \circ M) = (S \circ D) \circ (S \circ M)$ for an m-diagram S and an l-diagram M, such that $l > n > m > 0$.

The argument for $n > l > m$ is analogous. Hence, the statement $G(k)$ holds and the implication is true. ☐

Lemma 2.5.12. *For $k \geq 1$ the following statement holds: $H(k-1) \wedge G(k) \implies H(k)$, additionally H(0) holds with no further assumptions.*

Proof. Let us assume that both $H(k-1)$ and $G(k)$ hold.

For $H(k)$ to hold, we need to show that for any three well-defined diagrams: an n-diagrams D, an m-diagram S and an l-diagram M, such that $l, n > m > 0$ and $|l - n| = k$, the following holds:

$$\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M) = \mathrm{Inc}_r(S \circ D, S \circ M) \circ \mathrm{Inc}_r(S, M)$$

- For $k = 0$, we have $l = n > m$
- For $k = |l - n| > 0$, we have $l > n > m$ or $n > l > m$

First, assume $l \geq n > m$, to consider the cases $l = n > m$ and $l > n > m$ simultaneously. Since $l > n > 0$, to establish that these two embeddings are equivalent, by Lemma 2.3.6, we need to check the following conditions:

- Types are the same. By Definition 2.4.5 these types are as follows:

$$\begin{aligned}
&\mathrm{Inc}_r(D, M) : M \hookrightarrow D \circ M \\
&\mathrm{Inc}_r(S, M) : M \hookrightarrow S \circ M \\
&\mathrm{Inc}_r(S, D \circ M) : D \circ M \hookrightarrow S \circ (D \circ M) \\
&\mathrm{Inc}_r(S \circ D, S \circ M) : S \circ M \hookrightarrow (S \circ D) \circ (S \circ M)
\end{aligned}$$

  We could see that the domains are immediately equivalent, the codomains are equivalent by G(k)

- Heights are equal. For this condition, we consider two scenarios:

⋆ For $l = n > m$:

$$
\begin{aligned}
&(\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M)).h = \\
&\quad = (\mathrm{Inc}_r(S, D \circ M)).h + (\mathrm{Inc}_r(D, M)).h && \text{[Def. 2.3.2]} \\
&\quad = \mathrm{id}_{D \circ M}.h + |D| && \text{[Def. 2.4.5]} \\
&\quad = |D| && \text{[Def. 2.4.1]} \\
&\quad = |S \circ D| && \text{[Eq. (2.14)]} \\
&\quad = (\mathrm{Inc}_r(S \circ D, S \circ M)).h + (\mathrm{Inc}_r(S, M)).h && \text{[Def. 2.4.5]} \\
&\quad = (\mathrm{Inc}_r(S \circ D, S \circ M) \circ \mathrm{Inc}_r(S, M)).h && \text{[Def. 2.3.2]}
\end{aligned}
$$

⋆ For $l > n > m$:

$$
\begin{aligned}
&(\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M)).h = \\
&\quad = (\mathrm{Inc}_r(S, D \circ M)).h + (\mathrm{Inc}_r(D, M)).h && \text{[Def. 2.3.2]} \\
&\quad = \mathrm{id}_{D \circ M}.h + \mathrm{id}_M.h && \text{[Def. 2.4.5]} \\
&\quad = \mathrm{id}_M.h && \text{[Def. 2.4.1]} \\
&\quad = \mathrm{id}_{S \circ M}.h + \mathrm{id}_M.h && \text{[Def. 2.4.1]} \\
&\quad = (\mathrm{Inc}_r(S \circ D, S \circ M)).h + (\mathrm{Inc}_r(S, M)).h && \text{[Def. 2.4.5]} \\
&\quad = (\mathrm{Inc}_r(S \circ D, S \circ M) \circ \mathrm{Inc}_r(S, M)).h && \text{[Def. 2.3.2]}
\end{aligned}
$$

- Component embeddings are equivalent. Again, we consider two scenarios:

⋆ For $l = n > m$:

$$
\begin{aligned}
&(\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M)).e = \\
&\quad = (\mathrm{Inc}_r(S, D \circ M)).e.\Lambda[(D \circ M)[\mathrm{Inc}_r(D, M).h].d] \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.3.2]} \\
&\quad = (\mathrm{Inc}_r(S, D \circ M)).e.\Lambda[(D \circ M)[|D|].d] \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.4.5]} \\
&\quad = \mathrm{Inc}_r(S, (D \circ M)[|D|].d) \circ \mathrm{Inc}_r(D, M).e && [M(k)] \\
&\quad = \mathrm{Inc}_r(S, M[|D| - |D|].d) \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.4.7]} \\
&\quad = \mathrm{Inc}_r(S, M[0].d) \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.2.5]} \\
&\quad = \mathrm{Inc}_r(S, M.s) \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.2.5]} \\
&\quad = \mathrm{Inc}_r(S, M.s) \circ \mathrm{id}_M.e && \text{[Def. 2.4.5]} \\
&\quad = \mathrm{Inc}_r(S, M.s) \circ \mathrm{id}_{M.s} && \text{[Def. 2.4.1]} \\
&\quad = \mathrm{Inc}_r(S, M.s) && \text{[Def. 2.4.3]} \\
&\quad = \mathrm{id}_{S \circ M.s} \circ \mathrm{Inc}_r(S, M.s) && \text{[Def. 2.4.3]} \\
&\quad = \mathrm{id}_{S \circ M.s} \circ \mathrm{Inc}_r(S, M).e && \text{[Def. 2.4.5]} \\
&\quad = \mathrm{id}_{(S \circ M).s} \circ \mathrm{Inc}_r(S, M).e && \text{[Eq. (2.13)]} \\
&\quad = \mathrm{id}_{(S \circ M)[0].d} \circ \mathrm{Inc}_r(S, M).e && \text{[Def. 2.2.5]} \\
&\quad = (\mathrm{id}_{S \circ M}.e).\Lambda[(S \circ M)[0].d] \circ \mathrm{Inc}_r(S, M).e && [M(k)] \\
&\quad = ((\mathrm{Inc}_r(S \circ D, S \circ M)).e).\Lambda[(S \circ M)[0].d] \circ \mathrm{Inc}_r(S, M).e && \text{[Def. 2.4.1]} \\
&\quad = ((\mathrm{Inc}_r(S \circ D, S \circ M)).e).\Lambda[(S \circ M)[\mathrm{Inc}_r(S, M).h].d] \circ \mathrm{Inc}_r(S, M).e && \text{[Def. 2.4.5]} \\
&\quad = (\mathrm{Inc}_r(S \circ D, S \circ M) \circ \mathrm{Inc}_r(S, M)).e && \text{[Def. 2.3.2]}
\end{aligned}
$$

⋆ For $l > n > m$:

$$
\begin{aligned}
&(\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M)).e = \\
&\quad = (\mathrm{Inc}_r(S, D \circ M)).e.\Lambda[(D \circ M)[\mathrm{Inc}_r(D, M).h].d] \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.3.2]} \\
&\quad = (\mathrm{Inc}_r(S, D \circ M)).e.\Lambda[(D \circ M)[0].d] \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.4.5]} \\
&\quad = \mathrm{Inc}_r(S, (D \circ M)[0].d) \circ \mathrm{Inc}_r(D, M).e && [M(k)] \\
&\quad = \mathrm{Inc}_r(S, (D \circ M).s) \circ \mathrm{Inc}_r(D, M).e && \text{[Def. 2.2.5]} \\
&\quad = \mathrm{Inc}_r(S, (D \circ M.s)) \circ \mathrm{Inc}_r(D, M.s) && \text{[Eq. (2.13)]} \\
&\quad = \mathrm{Inc}_r(S \circ D, S \circ M.s) \circ \mathrm{Inc}_r(S, M.s) && [IH] \\
&\quad = \mathrm{Inc}_r(S \circ D, (S \circ M).s) \circ \mathrm{Inc}_r(S, M).e && \text{[Eq. (2.13)]} \\
&\quad = \mathrm{Inc}_r(S \circ D, (S \circ M)[0].d) \circ \mathrm{Inc}_r(S, M).e && \text{[Def. 2.2.5]} \\
&\quad = (\mathrm{Inc}_r(S \circ D, S \circ M)).e.\Lambda[(S \circ M)[0].d] \circ \mathrm{Inc}_r(S, M).e && [M(k)] \\
&\quad = (\mathrm{Inc}_r(S \circ D, S \circ M)).e.\Lambda[(S \circ M)[\mathrm{Inc}_r(S, M).h].d] \circ \mathrm{Inc}_r(S, M).e && \text{[Def. 2.4.5]} \\
&\quad = (\mathrm{Inc}_r(S \circ D, S \circ M) \circ \mathrm{Inc}_r(S, M)).e && \text{[Def. 2.3.2]}
\end{aligned}
$$

As all these conditions are satisfied both embeddings are equivalent. The argument for $n > l > m$ is analogous. By this, we established that $H(k)$ holds, hence the implication is true. Additionally, since we used the assumptions $G(k)$ and $H(k-1)$ only for $k > 0$, H(0) holds with no further assumptions. □

Finally, we can put all the pieces together to establish that the conjunction of G and H holds for all $k \geq 0$:

Lemma 2.5.13. *For $k \geq 0$: the following holds: $G(k) \wedge H(k)$*

Proof. We prove this by induction on k:

- *Base case*: For $k = 0$
  - ⋆ $G(0)$ holds by Lemma 2.5.10.
  - ⋆ $H(0)$ holds by Lemma 2.5.12.
- *Inductive step*: For $k > 0$ we assume that both $F(k-1)$ and $E(k-1)$ hold.
  - ⋆ To establish $G(k)$, by Lemma 2.5.11 we need both $G(k-1)$ and $H(k-1)$ to hold. Both hold by the inductive hypothesis.
  - ⋆ To establish $H(k)$, by Lemma 2.5.12 we need both $G(k)$ and $H(k-1)$ to hold. Both hold by the statement above.

By this inductive argument the statement $G(k) \wedge H(k)$ holds for $k \geq 0$.

□

Chapter 3

Application to quasistrict n-categories

In this chapter we formally introduce the graphical formalism for higher-dimensional diagrams. As stated before, it is a convenient notational shorthand, which in low dimensions can be made entirely rigorous by showing soundness and completeness results [**?**, 29, 55]. Introduction of this notation is consistent with the idea of finding the right level of abstraction when reasoning about mathematical structures. Here, we are able to absorb into the notation the details of individual generators and embeddings and how they fit together.

Later in Section 3.2, we utilise the graphical method of expression to explore the richness of higher-level coherences resulting from introduction of the interchange law. We analyse the interactions between different types of coherences and ordinary n-cells for $n = 2, 3, 4$. This lets us form definitions of quasistrict 2-, 3- and 4-categories expressed in the language of the diagram and signature structures defined in Chapter 2.

In the last part of the chapter in Sections 3.4 and 3.5, we discuss other attempts to define semistrict higher categories present in the literature: the 4-tas definition proposed by Crans [20] and the switch 3-category definition proposed by Douglas and Henriques [22]. We thoroughly compare these approaches with the definitions in this chapter and we contrast their treatment of strictness with ours. Finally, we show that a quasistrict 3-category defined according to Definition 3.2.2 satisfies the axioms of a switch 3-category and as a result also of a **Gray**-category. This allows us to retrieve some familiar categorical constructions in Section 3.6 and gives further evidence for correctness of this approach and the proposed new definition of a quasistrict 4-category.

3.1 Graphical formalism

In this section we introduce the graphical notation for diagram structures and work through an example where a graphical representation of a particular diagram is created. Subsequently, we discuss different types of composition in a quasistrict 3-category and present the graphical representations that accompany them. Finally, we provide intuition on how the graphical notation may be used for $n = 4, 5, 6$ when we run out of spatial dimensions in which different cells could be embedded.

Working with an n-dimensional structure required to satisfy multiple globularity conditions may be cumbersome for small n and quickly becomes unmanageable as n increases. One of the convenient tools that can be used in these instances is a graphical notation. The totality of equations governing such a structure is inherently difficult to comprehend, as multiple dimensions of the structure get squashed into a 1-dimensional line of an equation. A graphical calculus that makes use of geometrical structures can make this transition less drastic. This is especially the case for dimensions up to $n = 4$ where we have three spatial dimensions and one temporal dimension to utilise.

However, usefulness of the graphical notation also has its limits. This is the case, since for dimensions higher than four, human cognitive abilities make it difficult to visualise the geometrical structures that arise. The method that still allows us to gain some insight, even for dimensions $n > 4$, involves projecting out the lowest dimensions of the graphical representation, which are less likely to exhibit the properties that are of interest. That way, the images created are more easily comprehensible to humans.

An important remark to be made at this point is that throughout the reminder of this thesis we often use this graphical notation to reason about higher categories. Each time we do that however, we are not merely making intuitive, informal illustrations, we are in fact making fully formal statements about the underlying combinatorial representation depending on the signature and diagram structures.

First we provide a method of translating a diagram structure into its graphical representation. Intuitively, in an n-diagram D, each of its k-generators is depicted by an $n-k$ dimensional geometrical object. In particular, its n-generators are expressed as points, the generators in the source $s(D)$ as lines, generators in $s(s(D))$ as surfaces and so on. More formally, we propose a recursive definition of a graphical representation of a diagram.

Definition 3.1.1. For an n-diagram D, its *graphical representation* $G_D \subset \mathbb{R}^n$ is a labelled partitioned subspace, defined:

- For $n = 0$, to be $G_D = \mathbb{R}^0$
- For $n > 0$:
  - ⋆ at height i, to agree with $G_{D.[i].d} \subset \mathbb{R}^{n-1} \subset \mathbb{R}^n$;
  - ⋆ between heights, as a glued double cone modulo identifications.

This is not yet an entirely rigorous treatment of the matter, but we believe that it is sufficient for the purposes of depicting higher level cells in this chapter. We hope that a fully formal definition can be obtained by building on this concept. To illustrate the idea behind the definition, let us work through the following example step by step:

0-cells	A, B, C
1-cells	$F : A \to B$
	$G : B \to C$
	$H : C \to C$
2-cells	$\mu : H \Rightarrow G \circ H$
	$\nu : F \circ G \Rightarrow F$
	$\phi : H \Rightarrow \mathrm{id}_C$

The colouring is used to make the relation between the elements and their graphical representation more apparent. Then consider the following 2-diagram:

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

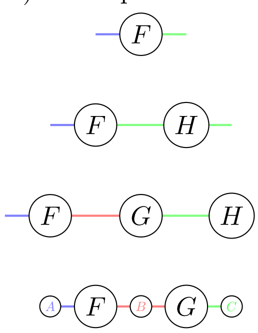
Note that according to Definition 2.2.7, embeddings between 0-diagrams consist of no data. For this reason the lists $(s(D))[0].e$ an$(s(D))[1].e$ are empty.

Given a 0-diagram S, we depict its sole 0-cell as a 0-dimensional object - a point. Given an n-diagram D, we first create $|S|$ pictures of its slices. Since they are themselves $(n-1)$-diagrams, this step is achieved recursively. For the given example, we first draw points for each 0-cell in the 0-diagram $s(s(D))$:

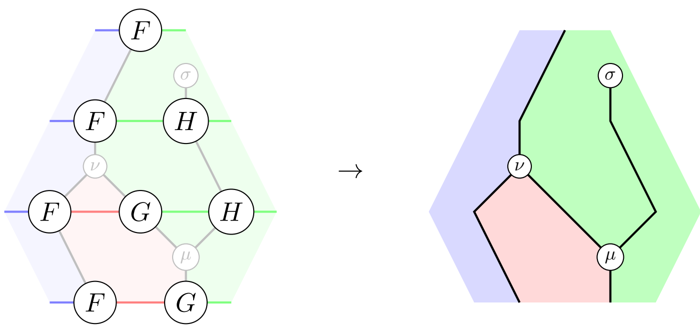
Then given an n-cell $S[i].g$ and slices $S[i].d$, $S[i+1].d$ that it connects, we do the following: first, for each geometric object representing a cell in either $S[i].d$ or $S[i+1].d$ we take its product with the unit interval $[0, 1]$. This increases the dimension of the representation and turns points into lines, lines into regions *etc.* Then, we connect both by creating a point corresponding to the n-cell $S[i].g$, we examine $s(S[i].g)$ and $t(D[i].g)$ as subdiagrams of $S[i].d$ and $S[i+1].d$ respectively and connect each $(n-1)$-cell involved in $S[i].g$ to this new point.

Any cell in $S[i].d$ which is not a part of $s(S[i].g)$ remains unaltered by the n-cell, hence is connected with its counterpart in $S[i+1].d$. The connection between the new point and the $(n-1)$-dimensional structures forms a *quasi* cone, since representations of $(n-2)$-cells must be extended to the boundaries of the picture as they are subject to globularity conditions.

We illustrate this by first turning the 0-cells in $s(s(D))$ into lines and then adding 1-cells in $s(D)$ to create the graphical representation of $s(D)$. We repeat this for all the other 1-slices of D:



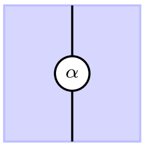
Then finally, we apply the recipe for combining $(n-1)$-slices into an n-diagram again, to obtain the following:

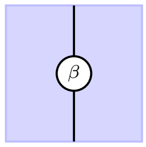


By discarding lowest-level coordinates we can project $\mathbb{R}^n \to \mathbb{R}^m$ for $m < n$. Additionally, given an n-diagram D, we could alternatively omit the final merge and represent D as a series of separate pictures of its slices: from $G_{s(D)}$ to $G_{t(D)}$. This corresponds to the idea of representing an n-dimensional structure by a series of $(n-1)$-dimensional snapshots. We use this feature often, especially for diagrams that consist of a single generator.

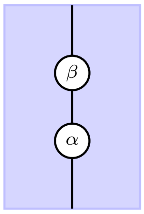
A direct consequence of this construction is the graphical interpretation of the procedures of taking an identity of a diagram and composing two diagrams, which are described in Definitions 2.4.6,2.4.22. Given a graphical representation G_d of the diagram D, the graphical representation of $\mathrm{Id}(D)$ is simply the product of G_d with the interval. For an n-diagram D and an m-diagram S that share a common boundary, the graphical representation of the composite $S \circ D$ is realised by pasting G_S and G_D along their shared boundary. If $m < n$, we need to apply the identity operation $n-m$ times on S first, so that the product of G_S with the interval is taken $n-m$ times. Then we paste the resulting representation and G_D along their shared boundary. The procedure is analogous for $n < m$.

Let us consider the example of two 3-diagrams S, D each consisting of a single generator. Their graphical representations are as follows:

$S =$ 

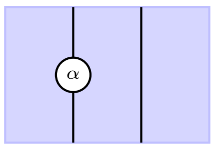
$D =$ 

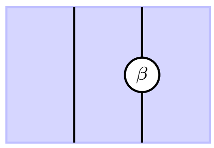
These are both 3D pictures, the 0-cells represented by spaces at the front and the back of the picture are not labelled. By Definition 2.4.6, there is only one way in which they could be directly composed. This is realised by diagram composition directly, so we consider the composite $S \circ D$. We take the graphical representations G_S and G_D and paste them together by placing G_D *on top* of G_S:



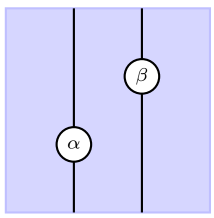
However, due to the whiskering perspective that we adapted, there are also two indirect methods of composition which correspond to horizontal and spatial composition. Given that there is a degree of choice on the order in which we compose the morphisms, we have multiple variants for each type of composition. These are as follows, assuming that appropriate sources and targets match and the composites exist:

- Horizontal composition:
  - ⋆ $(S \circ s(D)) \circ (t(S) \circ D)$: This corresponds to 1-composition of 3-cells in a 3-category. As the dimensions of S and $s(D)$ do not match, we create the graphical representation of $\mathrm{Id}(s(D))$ first before pasting with G_S, similarly for $t(S)$ and D. We obtain the following:

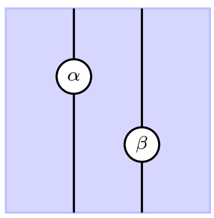
    $S \circ s(D) \quad =$ 

    $t(S) \circ D \quad =$ 

    Then since both $(S \circ s(D))$ and $(t(S) \circ D)$ are 3-diagrams, we can paste their graphical representations together to obtain:

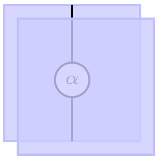
    

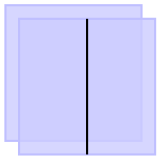
  - ⋆ $(s(S) \circ D) \circ (S \circ T(D))$: This is obtained analogously to the case above.

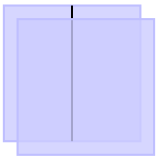
    

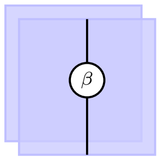
- Spatial composition:

⋆ $[(S \circ s(s(D))) \circ (t(s(S)) \circ s(D))] \circ [(t(S) \circ s(s(D))) \circ (t(t(S)) \circ D)]$: This corresponds to 0-composition of 3-cells in a 3-category. Similarly as with horizontal composition, we need to build up the graphical representations gradually:

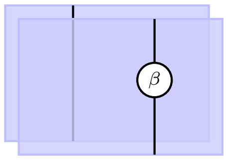
$S \circ t(t(D)) \quad =$ 

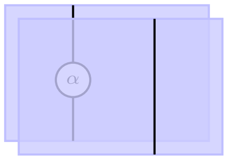
$t(s(S)) \circ s(D) \quad =$ 

$t(S) \circ s(s(D)) \quad =$ 

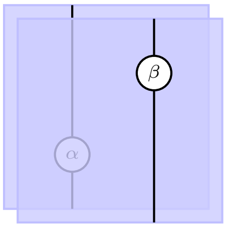
$t(t(S)) \circ D \quad =$ 

These could be combined to give:

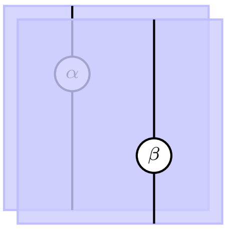
$(t(S) \circ s(s(D))) \circ (t(t(S)) \circ D) \quad =$ 

$(S \circ t(t(D))) \circ (t(s(S)) \circ s(D)) \quad =$ 

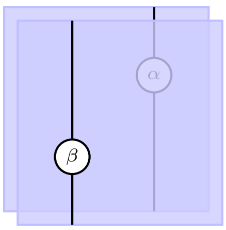
Then since both diagrams are of the same dimension, we finally paste these two together to obtain:



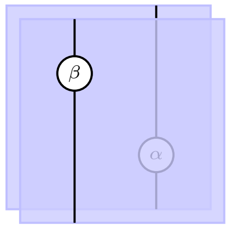
⋆ $[(s(S) \circ s(s(D))) \circ (t(t(S)) \circ D)] \circ [(S \circ s(t(D))) \circ (t(t(S)) \circ t(D))]$: This is obtained analogously to the case above.



⋆ $[(s(s(S)) \circ D) \circ (s(S) \circ t(t(D)))] \circ [(s(s(S)) \circ t(D)) \circ (S \circ t(t(D)))]$: This is obtained analogously to the first case.



⋆ $[(s(s(S)) \circ s(D)) \circ (S \circ s(t(D)))] \circ [(t(s(S)) \circ D) \circ (t(S) \circ t(t(D)))]$: This is obtained analogously to the first case.



For $n = 3$, we create 3D graphical representations and depict them on a 2D sheet of paper by creating the effect of transparency and a quasi 3D perspective of placing geometrical objects more in front or more at the back of the sheet. For $n = 4$ we run out of spatial dimensions, so it appears that this method does not give a significant expressive advantage over the purely algebraical representation. Even though we avoid having to squash multidimensional structures into an inherently one-dimensional equation. There is however a method of extending the operational usefulness of this approach, for $n = 4, 5, 6$.

Here we provide the intuition of how higher dimensional categorical structures could be expressed in the graphical formalism:

- For $n = 4$ imagine a 4-cell as a smooth transition between two 3D geometrical objects. A composite of multiple cells is then a “movie” of 3D snapshots, which could be thought of as a history of how the first image (scene) in the series got turned into the final one.
- For $n = 5$, a 5-cell can be thought of as a method of rewriting one movie (a 4D object) into another movie. There is however a requirement imposed by the globularity conditions that both movies must have the same first scene and the same final scene. A composite 5-cell is then a multi-step method of rewriting one movie into another with many intermediate movies in between.
- For $n = 6$, a 6-cell can be thought of as modifying one method of rewriting a movie, into another method. Imagine for instance that one method first rewrites some scenes in the initial part of the movie and then some non-overlapping scenes in the final part, then imagine that the second method does the opposite. Then these two methods of rewriting one movie of 3D objects into another, are related by a 6-cell.

For $n \geq 7$ it becomes increasingly difficult to picture in one's head what the representation of a cell in the graphical formalism is. This is however not an insurmountable obstacle to applicability of the diagram and signature formalism. This is for two reasons: Firstly, we could still in theory examine the underlying algebraic structures and gain some insight about their properties. Secondly, we could project out the lowest level dimensions that are of less interest and still explicitly depict up to six top dimensions using the graphical methods described above.

We conjecture that there are correctness and completeness results for the graphical calculus for the diagram and signature structures.

Conjecture 3.1.2. Graphical representations G_D, G_S for two k-diagrams D, S over a signature modelling a quasistrict n-category are the same if and only if the underlying diagram structures are equivalent.

We do not provide a rigorous proof here, however the outline is as follows:

- “⇒”: Should follow by the procedure of producing pictures outlined in Definition 3.1.1.
- “⇐”: An inductive argument on the diagram dimension n.

The graphical formalism can be used to prove various properties of the diagram and signature structures. Results such as associativity or distributivity of diagram composition (equations Eq. (2.21), Eq. (2.23), Eq. (2.22)) could now alternatively be proved in a graphical fashion. This method would require translating diagrams on both sides of the algebraic equality into their graphical representations and then comparing whether these representations are the same. This is what we mean by saying that certain low-level details get absorbed into the notation. For instance, associativity of diagram composition is implicit in the graphical notation.

3.2 Quasistrict 2-, 3- and 4-categories

In this section we explore the richness of higher level coherences that arise in quasistrict categories from the introduction of the interchange law. We present six types of interchanger morphisms and define what it means for a signature to *support* an interchanger of the given type. Throughout we use the graphical formalism described in the previous section.

In the quest to construct presentations of 2-, 3- and 4-categories using the diagram and signature formalism described in Chapter 2, we follow Gray's approach to semistrictness. The only non-trivial weakness that we allow is the interchange law and higher-level coherences that arise from it. We do not include any non-trivial associator or unitor morphisms. As shown in Section 2.5 these are already built-in within the diagram structure as strict equalities.

The final step towards the new definitions of quasistrict n-categories for $n = 2, 3, 4$ is to endow the signature structure with the notion of a non-trivial interchanger. In this section, we use the graphical formalism for diagrams to define six types of interchanger morphisms, which we denote using Roman numerals with a lower index k that indicates to what cell dimension the type is referring to. For example, I_2 denotes the interchanger of type I acting on 2-cells. Where appropriate, we additionally use informal names that correspond to the visual effect that the interchanger type has on the graphical representation of the diagram it acts on. We also discuss application of interchanger morphisms to composite cells and justify why we do not include explicit expansion morphisms between composite and atomic interchangers.

As discussed in Section 2.1, defining a category by providing its presentation is an entirely valid and complete approach. Here, we structure the definitions in the form of a presentation by the n-polygraph-like signature structure that is additionally endowed with the appropriate interchanger morphisms. We list these definitions immediately and define the different interchanger types in the later part of the chapter.

Definition 3.2.1. A *quasistrict 2-category* is a 3-signature that supports interchangers of type:

- 3-cell interchangers: I_2

Definition 3.2.2. A *quasistrict 3-category* is a 4-signature that supports interchangers of types:

- 3-cell interchangers: I_2
- 4-cell interchangers: I_3, II_3

Definition 3.2.3. A *quasistrict 4-category* is a 5-signature that supports interchangers of types:

- 3-cell interchangers: I_2
- 4-cell interchangers: I_3, II_3
- 5-cell interchangers: I_4, II_4, III_4, IV_4, V_4 and VI_4.

These definitions are much different than the standard approach to defining higher categories. The most significant divergence is the lack of conditions that the individual maps are required to satisfy. It is however the case that, because of the underlying diagram and signature structures, many properties are built-in and instead hold implicitly. This includes associativity, unitality and distributivity of composition, as well as different conditions normally imposed on the behaviour of interchanger morphisms. The intricacies of all these rules are still captured, but are now hidden behind the simple notions of diagram and signature structures.

In our setup, given an $(n+1)$-signature $\sigma = \{G_0, ..., G_{n+1}\}$, for each $0 \leq k \leq n$, we have the following. Firstly, the elements of each set G_k are the generating k-cells for the category. Secondly, the sets ${\Delta^*}_k$ of all k-diagrams over σ are the sets of all composite k-cells in the category. Finally, the elements in the set G_{n+1} are relations between n-cells, or in other words, conditions that the n-cells in the n-category satisfy. For these reasons, in the reminder of this thesis we use the notions of k-diagrams and composite k-cells interchangeably.

In a weak n-category for $2 \leq k \leq n$, every pair of k-morphisms f, g that is $(k-2)$-composed, gives rise to an interchanger morphism, which in turn gives rise to higher-level coherences for $k < n$. Even though for the purpose of defining a quasistrict 4-category, we only need specific interchanger k-morphisms up to $k = 5$, in the following exposition we define different interchanger types in their full generality for any $k \leq n$. The main benefit is that this allows us to discuss some higher-level coherences that quasistrict n-categories need to satisfy for $n \geq 5$. This is by no means sufficient to define them, but it is a step in the right direction. In Definitions 3.2.1, 3.2.2 and 3.2.3, we instantiate these general interchanger types to the particular dimensions that play a role in the given quasistrict n-category.

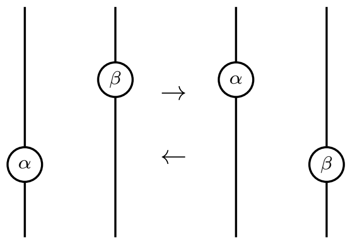
3.2.1 Interchange law

We begin the exposition by discussing the interchange law which is the source of all higher level coherences that arise in quasistrict n-categories. Informally, in a weak 2-category, the interchange law (also referred to in the literature as the 'exchange law') for 2-cells states that there is an equivalence between the two different orders of horizontal composition for consecutive n-cells. Formally this is expressed as the following, provided that these composites exist:

Definition 3.2.4. In a weak 2-category the *interchange law* states that for two 2-cells α and β there is the following equivalence:

$$(\alpha \circ s(\beta)) \circ (t(\alpha) \circ \beta) \simeq (s(\alpha) \circ \beta) \circ (\alpha \circ t(\beta))$$

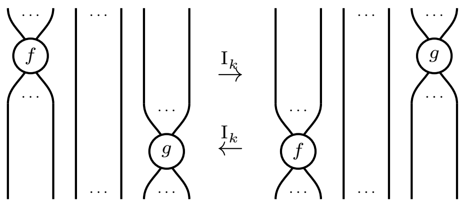
It is graphically represented by the following 3-cells:



We could see that composition via whiskering gives us two alternative methods of horizontally composing two 2-cells, which are related by a 3-cell. More generally, the presence of interchanger morphisms means that the diagrams resulting from these alternative methods of composition are related by higher level morphisms, *i.e* there exists a series of rewrites turning one representation into the other and vice versa.

Recall that in an n-signature $\sigma = (G_0, ..., G_n)$, for $0 \leq k \leq n$, each of the sets G_k consists of the generating (*i.e.* non-composite) cells. That way, cells $f, g \in G_k$ in are generators, not composite cells. We then formalise the generalised interchange law for k-cells in an $(n+1)$-signature as interchangers of type I in the following way:

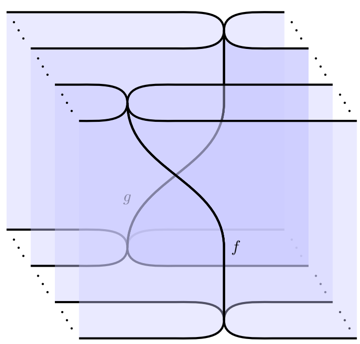
Definition 3.2.5. An $(n+1)$-signature $\sigma = (G_0, ..., G_{n+1})$ supports *interchangers of type* I_k such that $2 \leq k \leq n$, if for any k-cells $f, g \in G_k$, and for any $(k-1)$-cells $w_1, ..., w_m$, the following invertible cells are elements of the set G_{k+1}:



Interchangers of type I_k are $(k+1)$-cells that act on k-cells. In this graphical representation, the lowest $k-2$ dimensions have been projected out. This results in two 2D snapshots of the source and of the target of each interchanger. Each snapshot preserves all the features that are of interest in this context. The source and target of both k-cells f, g in the picture could be composed of multiple individual $(k-1)$-cells. There could also be an arbitrary number of $(k-1)$-cells in between the two k-cells. These two eventualities are accounted for by the presence of triple dots, which are superficial and not a part of the graphical formalism.

Note that the graphical representation of the interchange law gives us an intuitive condition on when two cells can be interchanged. If they are directly linked by a shared source to target connection, the interchange is not possible. Since the effect of this interchanger on the graphical representation is that the heights of the two cells involved are swapped, we also refer to this type of interchanger as a 'swap' morphism.

In accordance with the convention for creating pictures of diagrams in the graphical calculus outlined in Section 3.1, there is an alternative graphical expression for these morphisms. Instead of two 2D snapshots for the source and the target of the morphism separately, we express the entire 3-cell as a single 3D picture. In this view the braiding that arises is more visible:



In principle, interchangers of type I, and of all the other types introduced later, could be described in purely algebraic terms, without making any reference to the graphical formalism. However, that would make us lose all the potential benefits with regards to the clarity of presentation and absorption of low-level details into the formalism.

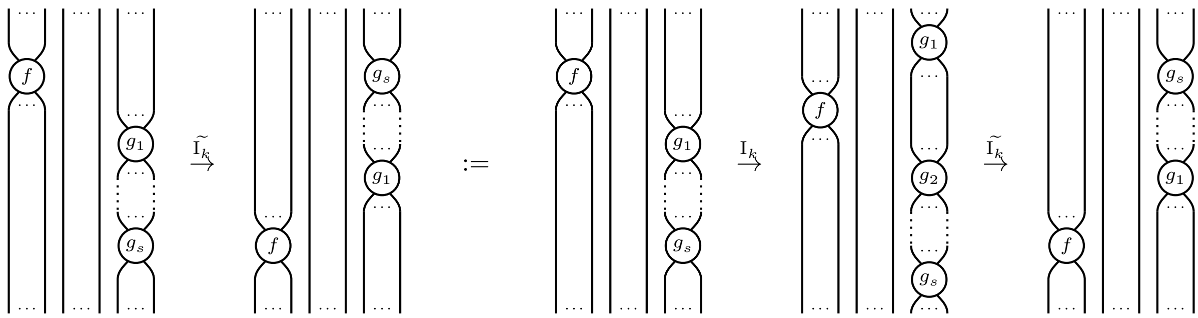
In the following exposition, we strive to make the graphical representations as general as possible, but at the same time to put emphasis on the most crucial features and not to obscure them with too much irrelevant detail. For this reason, from now on, for the sake of clarity of presentation, we omit the multiple $(k-2)$-cells in the sources and targets of $(k-1)$-cells, which are represented by additional sheets in the picture above. They do not contribute towards the general behaviour of higher-level coherences and, at the same time, they do obscure the more important features of the interchanger.

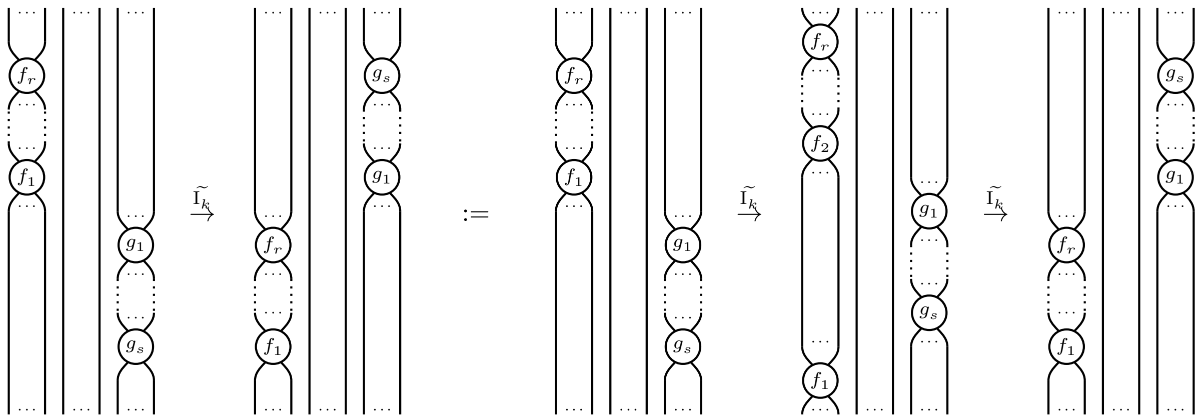
Up to this point, we only discussed the interchange law for two non-composite k-cells. For interchangers of type I and all the other types defined in this chapter, we distinguish between interchangers that act on composite k-cells and those that act on simple (non-composite) k-cells. We refer to the former as *composite* interchangers and to the latter as *atomic* interchangers. The key observation is that composite interchangers can always be realised as composites of atomic interchangers and, as such, do not have to be explicitly added to the signature.

The motivation for formalising composite interchanger is that, in the analysis of higher level coherences interchangers cells are themselves subject to higher-level relations. Hence, composite interchangers are necessary to define these higher level singularities in full generality.

In the graphical formalism each individual k-cell α_i that constitutes a part of the composite k-cell α is clearly distinguishable as a separate geometrical object. As a result, there is an intuitive concept of defining an interchanger morphism of type I acting on composite cells, in terms of a series of applications of the same (atomic) interchanger to individual non-composite cells.

Definition 3.2.6. In an $(n+1)$-signature that supports interchangers of type I_k for $1 < k \leq n$, a *composite interchanger* $\widetilde{\mathrm{I}_k}$ consists of a sequence of individual applications of the atomic interchanger of type I_k, with values defined recursively as follows:





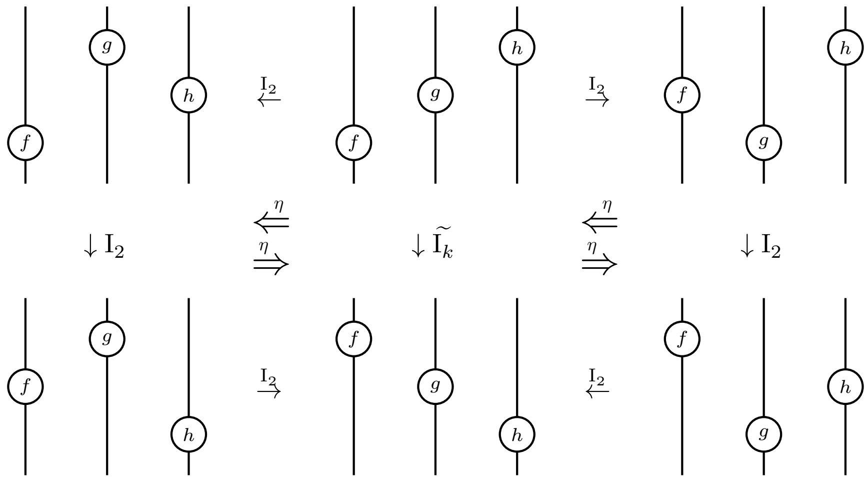
The intuitive interpretation is that when faced with interchanging composite cells, we interchange them one generator at the time. First interchanging the cell f_1 with all cells g_i, then repeating the procedure for each consecutive f_j.

This notion does not add any new structure to the signature supporting I_k, since $\widetilde{\mathrm{I}_k}$ is a composite of interchangers that are already present there. In this sense, $\widetilde{\mathrm{I}_k}$ is merely a notational shorthand that makes reasoning about interchanging composite k-cells more simple.

An important point to be discussed is why not to include composite interchangers as native cells and then introduce explicit *expansion* morphisms that would relate them to applications of atomic interchangers, as described above. This would certainly be in direct correspondence to axioms included in other approaches to defining semistrict categories (see axiom **S2-17** in 3.5.1). However both inclusion and exclusion of explicit composite interchangers and expansion morphisms are valid approaches and both would result in functioning definitions.

We acknowledge that the inclusion of these cells would result in a more weak definition of a category, however this increased weakness would not bring any additional expressivity. This is because any composite interchanger is already expressible as a composite of atomic interchanger applications. Moreover, not keeping this additional weakness is consistent with the spirit of the semistrict approach, which aims to obtain definitions of categories which are 'as strict as possible', while retaining equivalence to general weak n-categories.

There is also a more practical reason justifying our approach. Explicit expansion morphisms would give rise to a large amount of extra structure, again not bringing any additional expressivity. For instance, there is more than one possible way of applying n-cell expansion morphisms to unpack a contracted bundle of interchangers:



Both methods of expansion η would have to be related by a coherence of a new type: an 'expansionator', which for this example would be a 5-cell. Additionally, an expansion morphism could come into interaction with other singularities, leading to a further growth of the number of higher-level coherences. This would also impact on the length of the proofs conducted within the formalism. One of the primary reasons for following the quasistrict approach is to make proofs more manageable and the constant need to expand and contract composite interchangers could interfere with this goal.

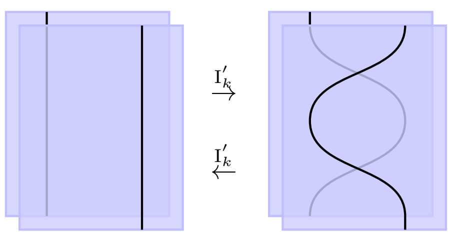
Before we proceed to defining the general interchanger types for higher categories, we need to first formally define what it means for a cell in a signature to be invertible. Since we are aiming to model quasistrict n-categories that are designed to retain equivalence to a general weak n-category, we interpret invertibility in the maximally weak sense.

Definition 3.2.7. Given a signature $\sigma = (G_0, ..., G_n)$, for $0 \leq k \leq n$ a k-cell $A \xrightarrow{f} B \in G_k$ is *invertible* if there exists a k-cell $B \xrightarrow{g} A \in G_k$ such that:

- For $k = n$, $f \circ g = \mathrm{id}_B$ and $g \circ f = \mathrm{id}_A$.
- For $k < n$, there exists two invertible $(k+1)$-cells f_1' and f_2' such that $f \circ g \xrightarrow{f_1'} \mathrm{id}_B$ and $\mathrm{id}_A \xrightarrow{f_2'} g \circ f$

We then say that g is the inverse of f and also refer to it as f^{-1}.

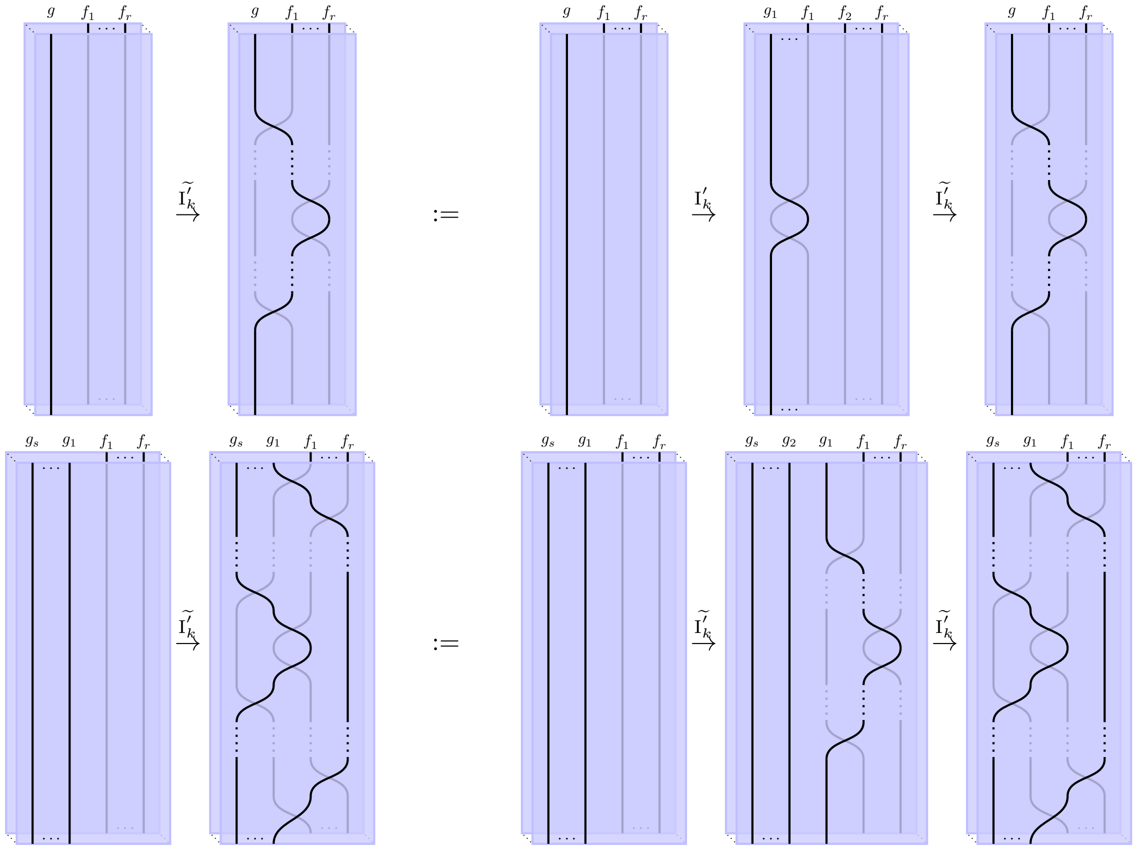
In the reminder of this thesis, whenever we refer to a cell being invertible, we always mean it in the sense of this definition. We also use the convention of denoting the higher level invertibility morphisms for interchangers by a prime. Then, for an interchanger of I_k, which is a $(k+1)$-cell, invertibility $(k+2)$-cells are denoted by I_k', invertibility $(k+3)$-cells by I_k'' *etc.* These cells can also be expressed graphically, an example is:



We informally refer to these morphisms as 'tangles' due to the geometrical effect they have on the wires in the graphical representation.

Similarly as for interchangers of type I_k, these invertibility cells could also act on composite cells. This is captured by the following definition:

Definition 3.2.8. In an $(n+1)$-signature that supports interchangers of type I_k for $1 < k < n$, a *composite invertibility cell* $\widetilde{\mathrm{I}'_k}$ consists of a sequence of individual applications of an individual cell of type I'_k, with values defined recursively as follows:



Similarly as with composite interchangers of type $\widetilde{\mathrm{I}_k}$, the intuitive interpretation is that we introduce or delete pairs of inverse interchangers of type I_k one pair at the time. First tangling the cell g_1 with all cells f_i, then repeating the procedure for each consecutive g_j. Again, for the same reasons as $\widetilde{\mathrm{I}_k}$, this notion does not add any new structure to the signature supporting I_k.

3.2.2 Higher-level coherences for $n = 3$

A crucial observation is that interchangers of type I_k give rise to higher-level coherences. This occurs when they are considered as $(k+1)$-morphisms in a quasistrict n-category for $k+1 \leq n$. In exploring the various higher-level coherences for each dimension, it is important to be certain that one has already found all that arise. One framework within which one may approach this is to exhaustively consider all the interactions between the interchanger cells and all the other types of k-cells in the category. For example, for $k = n-1$, we have the following:

-	n-cell	I_{n-1}
n-cell	I_n	II_n
I_{n-1}	-	$\mathrm{I}'_{n-1})$

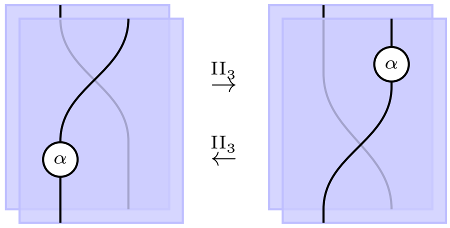
In particular, for $n = 3$ this gives us:

- I_2 interpreted graphically as crossings of wires on different sheets.
- I_3 to be thought of as the exchange of heights of two 3-cells in a 3D picture.
- I'_2, which are invertibility morphisms for type I_2, as defined in 3.2.7. These morphisms introduce or delete adjacent pairs of inverse interchangers of type I_2.

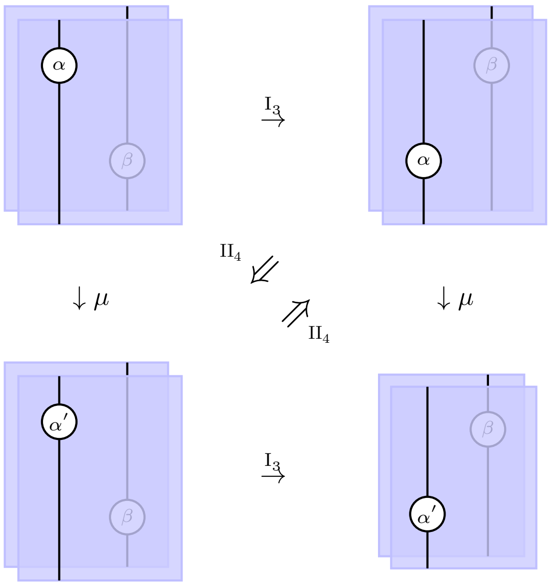
- An interchanger of new type II that arises when a 3-cell could be 'pulled-through' a crossing created by an interchanger of type I_2.

Cells of types I_2, I_3 and I'_2 have already been discussed in detail in the previous Section, here we concentrate on the higher-level coherence formalised as an interchanger of type II.

Interchangers of type II express the requirement that an interchanger of type I is natural in one variable. Interchangers of type II arise first one dimension above those of type I. This is because, in order to arise, II_k needs an instance of I_{k-1} to share its input with a k-cell. This is easiest understood through the example of a 3-cell α interacting with an instance of an interchanger of type I_2, which here is graphically expressed as a crossing of wires:



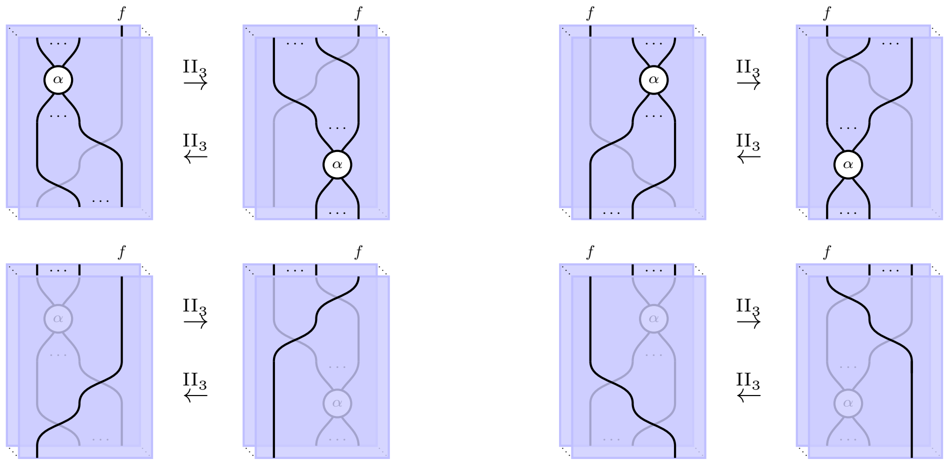
Because of the apparent motion of α that is being pulled through the crossing, we informally refer to interchangers of type II_k as 'pull-throughs'. Again, the integer k indicates the dimension of the cells that are being acted upon by this interchanger. Below, we provide intuition for the graphical representation of the most basic instance of II_4. For a 4-cell $\mu : \alpha \to \alpha'$ and a 3-cell β, the graphical representation of μ is by two 3D snapshots of its source and its target. This could be thought of as evolution of a 3D picture in time. The graphical interpretation of II_4 is as follows:



Here, we have a 4-cell μ that gets pulled through a crossing created in four dimensions by the swap of heights of two 3-cells α and β, *i.e.* the operation μ gets executed either *before* or *after* the heights of α and β are swapped.

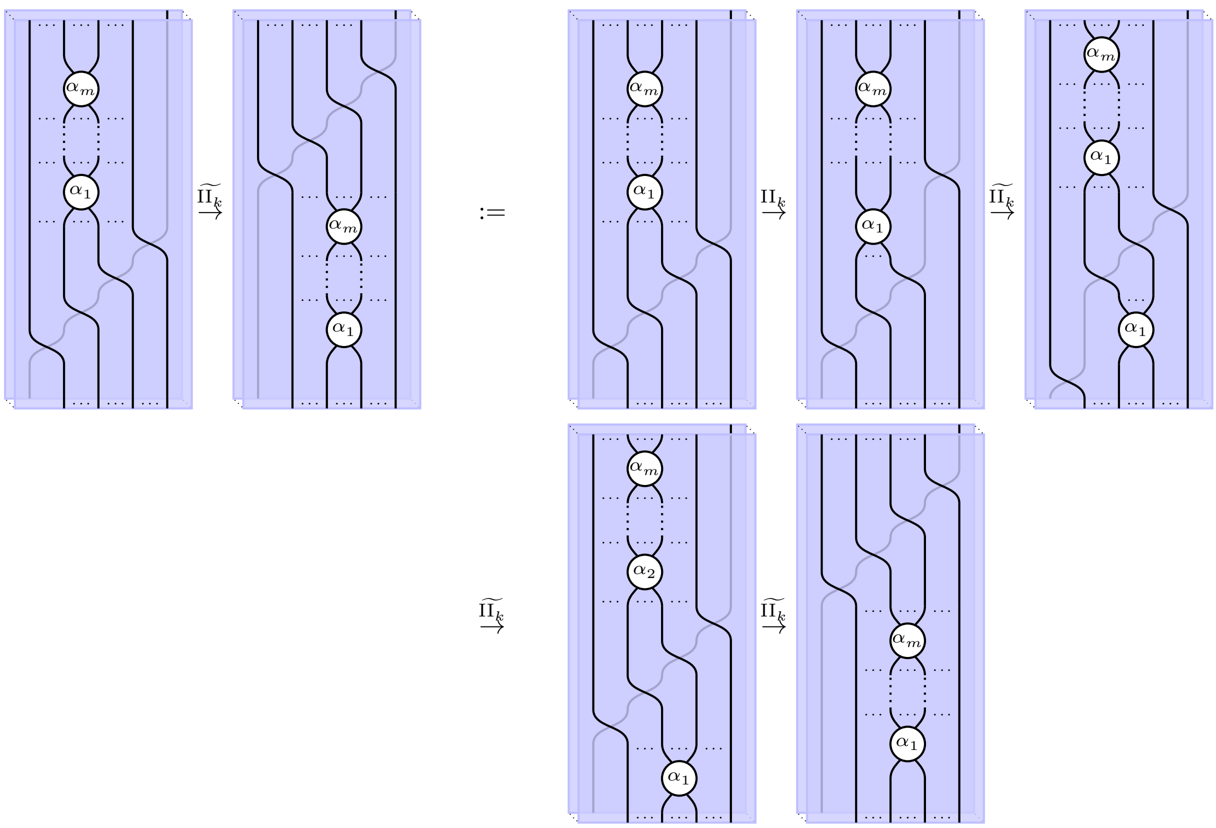
In full generality singularity of type II_k is defined in the following way:

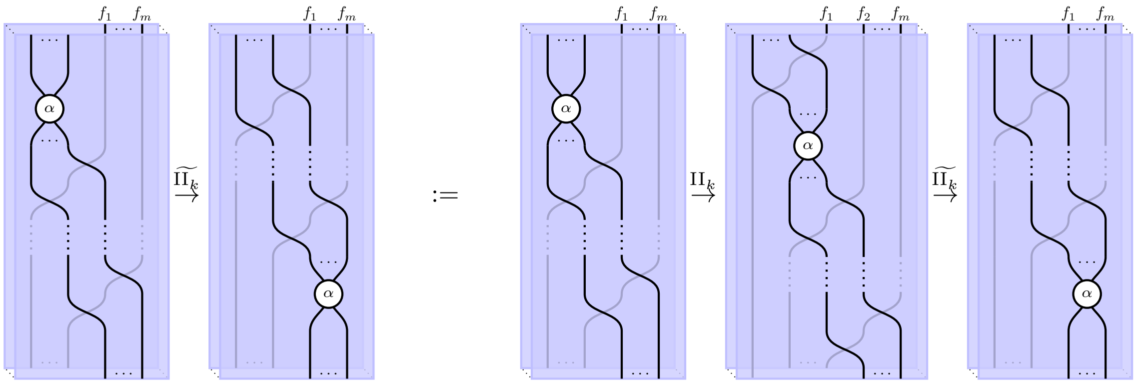
Definition 3.2.9. An $(n+1)$-signature $\sigma = (G_0, ..., G_{n+1})$ supports *interchangers of type* II_k such that $3 \le k \le n$, if for any k-cell $\alpha \in G_k$ and any $(k-1)$-cells$f$, the following invertible cells are elements of the set G_{k+1}:



Similarly as other singularity types discussed, interchangers of type II could also be applied to composite cells. According to the definition, we require a 3-cell α and a 2-cell f for the interchanger of type II to arise. This gives rise two variants of the composite interchanger, as either α or f could be a composite cell. We overload the notation and refer to both as $\widetilde{\mathrm{II}}_k$, in the reminder of this thesis the specific variant will always be clear from context.

Definition 3.2.10. In an $(n+1)$-signature that supports interchangers of type II_k for $1 < k \leq n$, a *composite interchanger* $\widetilde{\mathrm{II}}'_k$ consists of a sequence of individual applications of an individual cell of type II_k, with values defined recursively as follows:





Similarly as for other composite interchangers, the intuitive interpretation for $\widetilde{\mathrm{II}_k}$ is that we could pull a composite cell α through a single I_{k-1} crossing by pulling-through each non-composite cell α_i separately. Also, for pulling through multiple crossings, we could pull-through one crossing at a time. Again, for the same reasons as for $\widetilde{\mathrm{I}_k}$, the composite interchangers of type $\widetilde{\mathrm{II}_k}$ do not add any new structure to the signature supporting II_k.

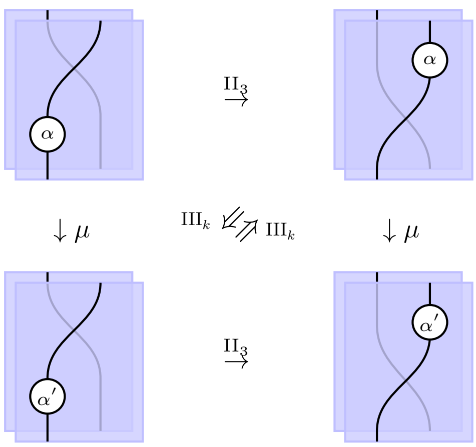
3.2.3 Higher-level coherences for $n = 4$

We proceed to exploring coherences that the interchange law for k-cells gives rise to at level $n = k + 2$. All singularities defined in the previous subsection are now subject to higher-level coherences themselves. Following the approach of investigating the interactions between different types of cells, we have the following table for $n = k + 2$ in a quasistrict n-category:

-	n-cell	I_{n-1}	I'_{n-2}	II_{n-2}
n-cell	I_n	II_n	IV_n	III_n
I_{n-1}	-	I'_{n-1}	-	V_n
I'_{n-2}	-	-	I''_{n-2}	-
II_{n-1}	-	-	-	II'_{n-2}

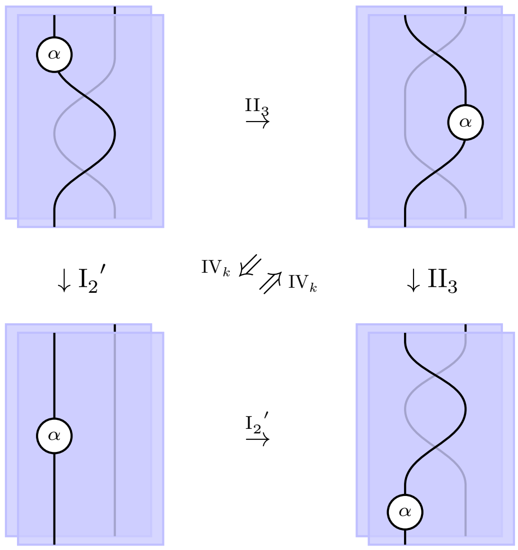
The higher-dimensional instances of interchangers of types I and II are covered by the general definitions presented in previous sections. The behaviour of new invertibility cells I', II' and II'_{n-2} is determined by Definition 3.2.7. Below, we discuss the three new types III-V arising, as well as a new type VI, which does not result from the interactions described in this table. We conclude this section with definitions for all these interchanger types in their most general forms.

Interchangers of type III express the requirement that an interchanger of type II is natural in one variable. Type III arises first when a 4-cell interacts with a crossing created by an interchanger of type I_2. Similarly as for II_4, the cell α could be turned into α' either before it is pulled-through the crossing or after.



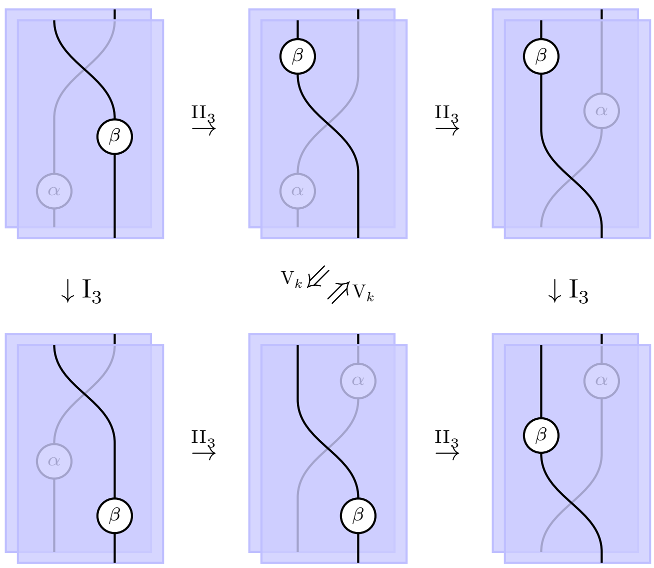
The intuitive interpretation here is that a 4-cell is being pulled through a crossing of 2-cells. An additional comment to be made is that for $n > 4$, at every level n, there is an analogous coherence where an n-cell gets pulled through a crossing of 2-cells.

Interchangers of type IV express the requirement that invertibility cells I' for interchangers of type I are natural in one variable. They arise first when an interchanger of type I_3 interacts with the invertibility 4-cell for interchangers of type I_2.The simplest instance is:



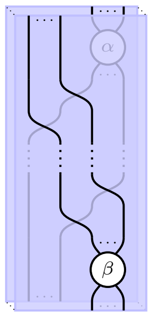
Intuitively, given tangled wires and a 3-cell, we could either pull the 3-cell through all the crossings or alternatively untangle all the wires and the tangle them again, this time above the 3-cell.

Interchangers of type V express the requirement that an interchanger of type I is natural in both of its variables simultaneously. They arise first when interchangers of types I_3 and II_3 interact with each other. Graphically, the interpretation is that, given a pair of adjacent 3-cells that do not share inputs or outputs, but whose inputs and outputs fully cross, there are different ways in which such a pair of 3-cells could be pulled through the crossing. The simplest instance is:



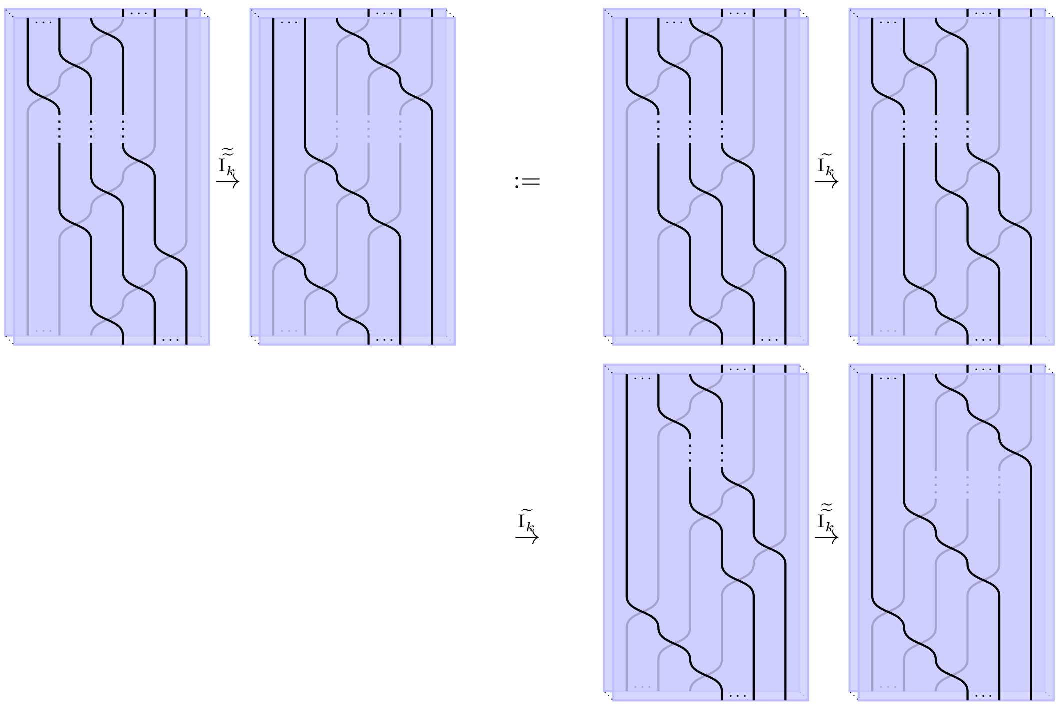
In these sequences of 4-cells, we swap the heights of 3-cells α and β using an instance of an interchanger of type I_3 either before or after they are pulled-through the crossing. Note that, the instances of I_3 used in both sequences are inverses.

In the most general form, both 3-cells could have composite 2-cells as its inputs and outputs, leading to multiple wires crossing. In that case, the additional difficulty is that after one 3-cell is pulled through, the crossings have to be readjusted, so that they are in the right order for the second 3-cell to be pulled through. In the picture below, β could be pulled *up*, but α cannot be pulled *down* because the crossings on the top wire below α are not adjacent:



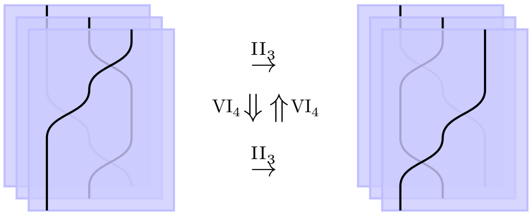
To counteract this, we need to account for reorganisation of crossings. This could be achieved by combining instances of the composite interchangers of type I, as defined in 3.2.6. Given a net of crossings resulting from interchanging two composite cells, we need to put them in the correct position, so that the interchanger of type II could be applied. We use $\widetilde{\widetilde{\mathrm{I}}}_k$ to denote the composite interchanger that achieves this.

Definition 3.2.11. In an $(n+1)$-signature σ that supports interchangers of type I_k, a *composite interchanger* $\widetilde{\widetilde{\mathrm{I}}}_k$ consists of individual applications of the composite interchanger of type $\widetilde{\mathrm{I}}_k$, with values defined recursively as follows:



The intuitive interpretation of this operation is that the first row of crossings gets put in their correct position first, then the procedure is recursively repeated with one fewer row of crossings to be organised. Again, as is the case with $\widetilde{\mathrm{II}}_k$, the cell $\widetilde{\widetilde{\mathrm{I}}}_k$ is just a convenient notational shorthand for multiple applications of the atomic interchanger of type I_k and, as such, does not add any additional new structure to the signature σ. Since σ supports interchangers of type I_k, all cells $\widetilde{\mathrm{II}}_k$, which are composites of atomic interchangers of type I_k, are already present. This extends to $\widetilde{\widetilde{\mathrm{I}}}_k$ as a composite of cells of type $\widetilde{\mathrm{II}}_k$.

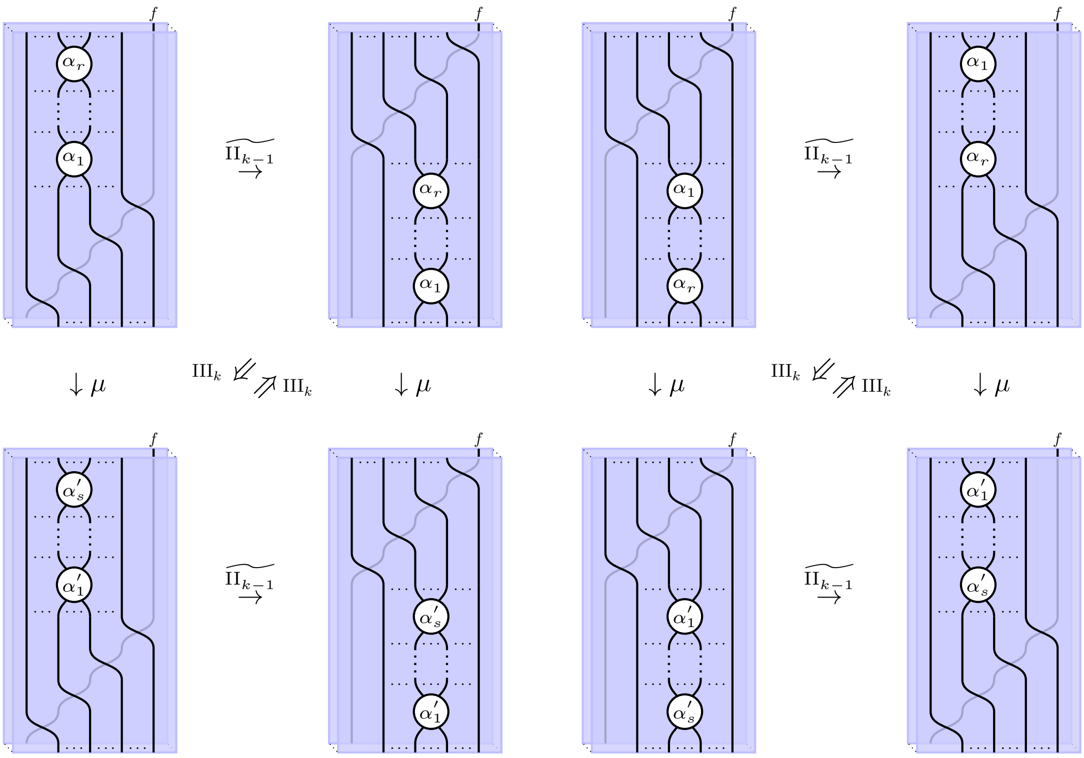
Interchanger of type VI is the final higher-level coherence that arises at this level. It is a morphism between two applications of an interchanger of type II, when the subject is an interchanger of type I. For II_3 this is visualised as follows:

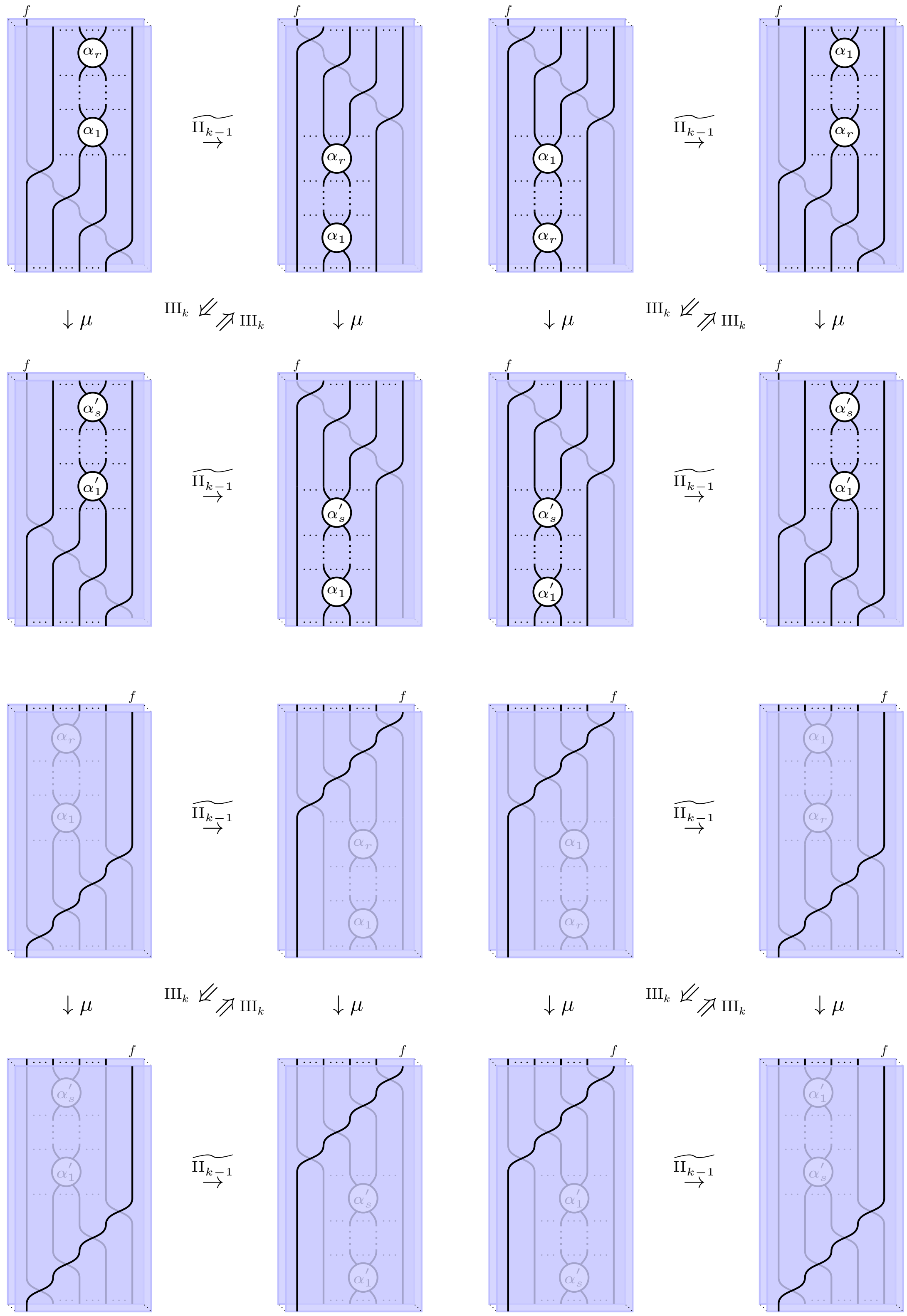


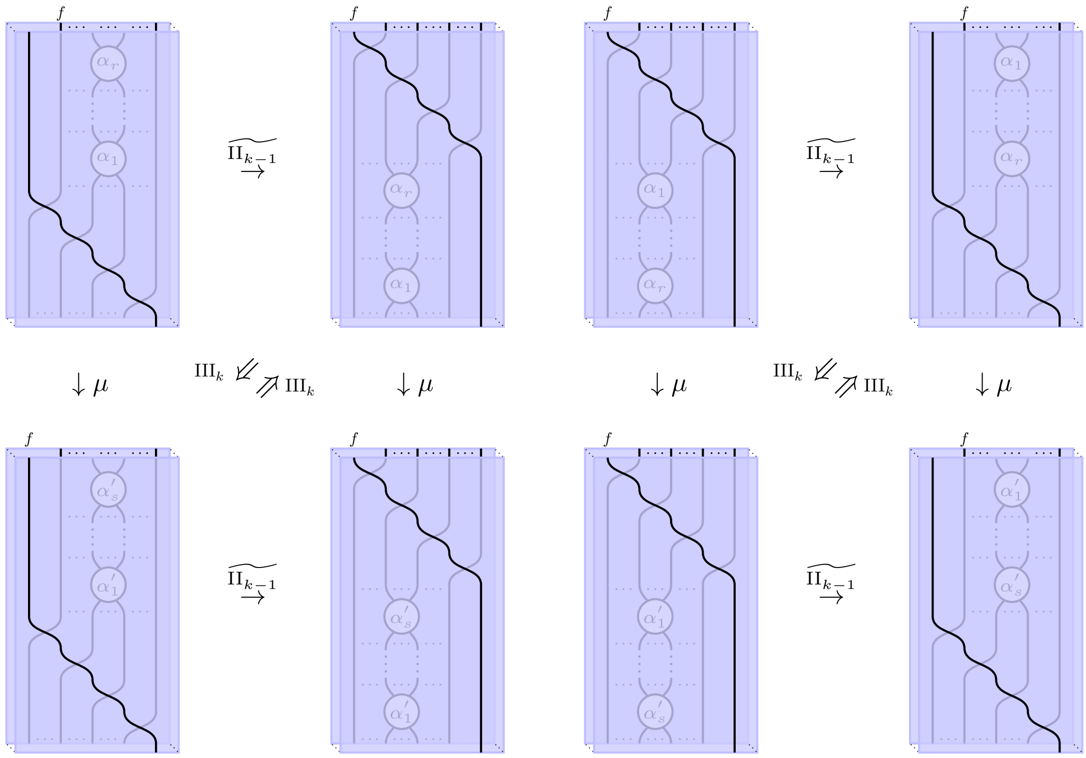
This special case of the interchanger of type II is exactly the third Reidemeister move. Looking at the left hand side first, the bottom left crossing gets pulled up through the two intermediate crossings and ends up at the top right corner. However, an equivalent interpretation is that the top left crossings gets pulled down to the bottom right. These two interpretations correspond to two different variants of the type II_3 interchanger applied at different heights, which therefore have to be related by a higher-level morphism. This observation was first made by Breen [12] and the cell is referred to in the literature as the 'Breenator'.

We conclude this section by providing definitions for interchangers of types III-VI in their most general form:

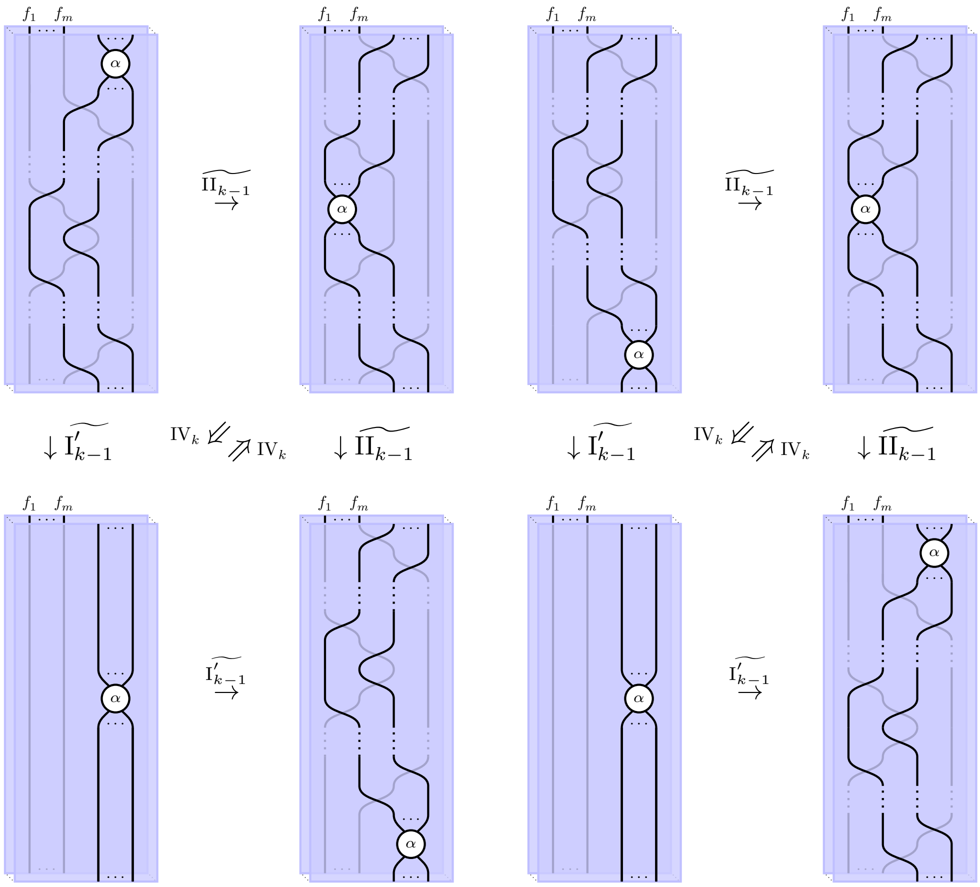
Definition 3.2.12. An $(n+1)$-signature $\sigma = (G_0, ..., G_{n+1})$ supports *interchangers of type* III_k such that $4 \leq k \leq n$, if for any k-cell $\mu \in G_k$ such that $\mu : \alpha \to \alpha'$ and any $(k-2)$-cell $f \in G_{k-2}$, the following invertible cells are elements of the set G_{k+1}:

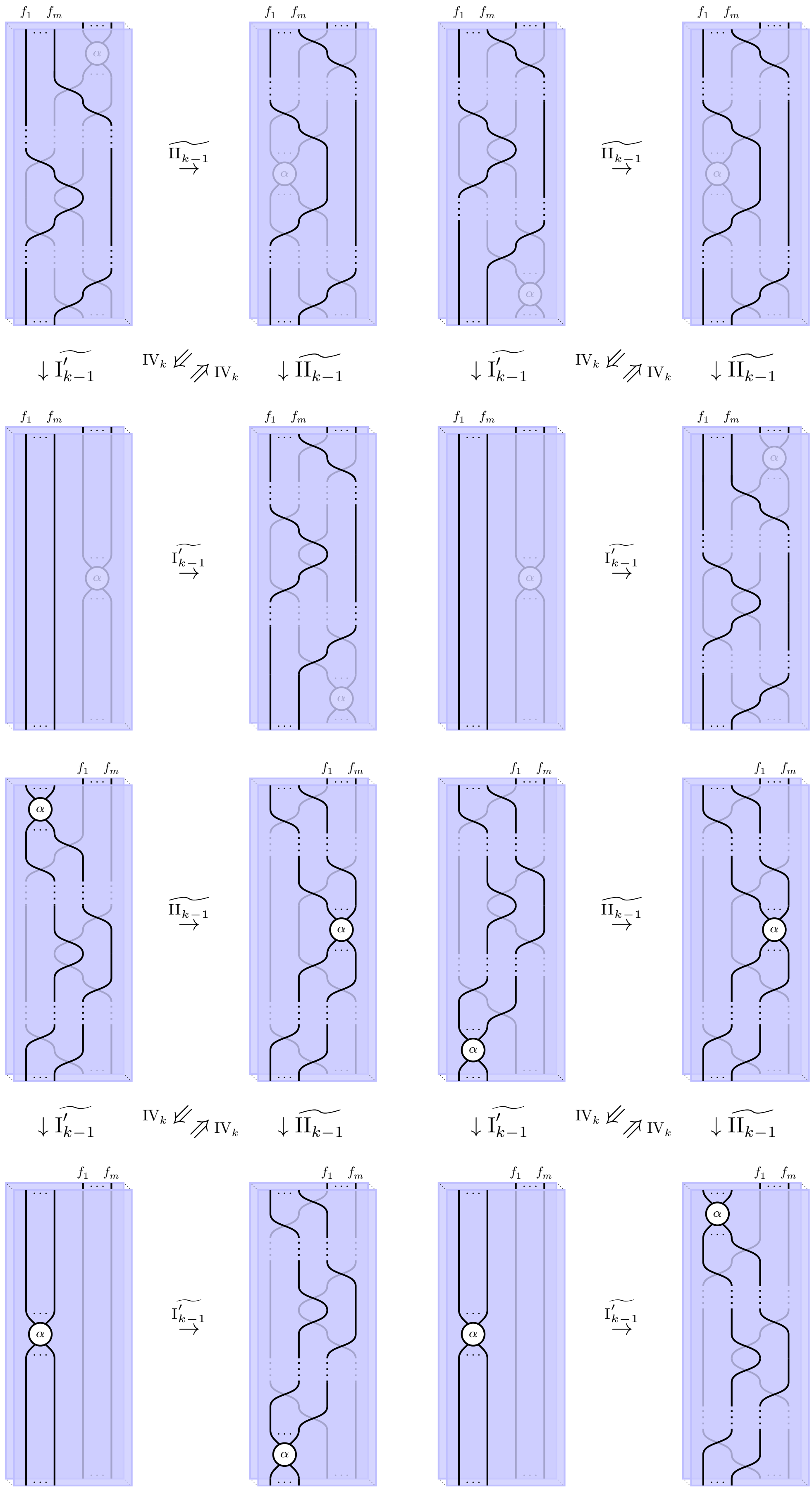


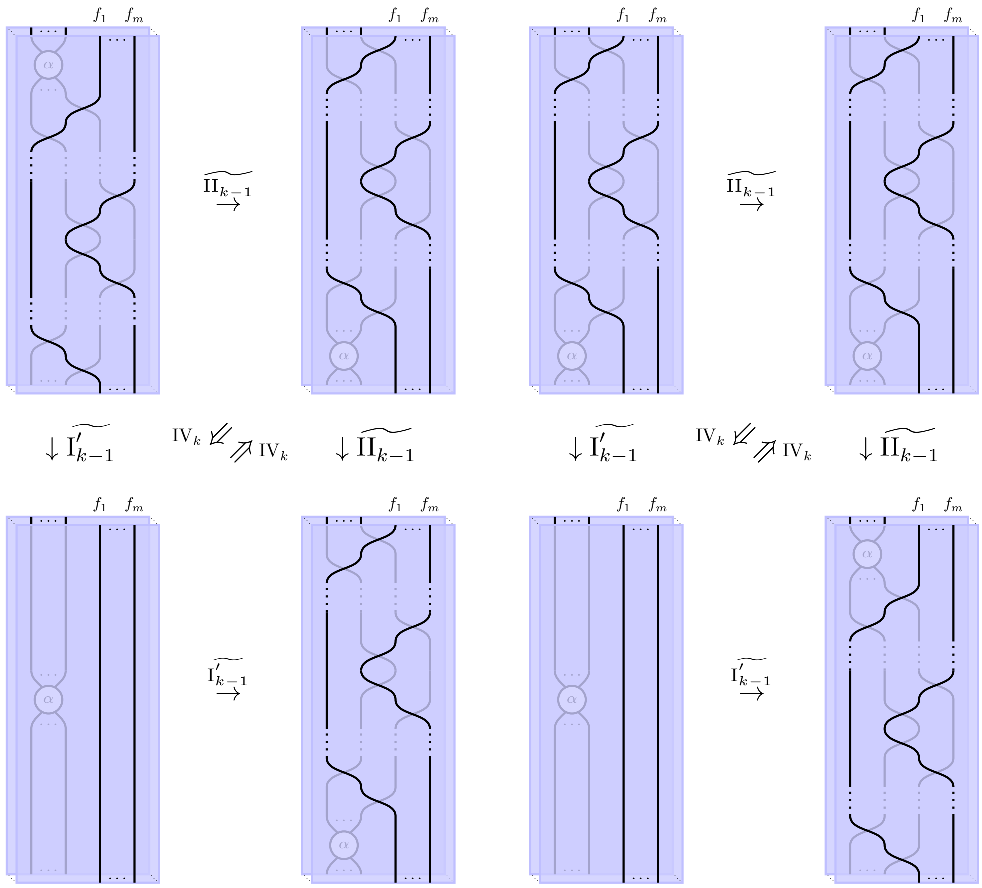

f
α_r
α_1
α'_s
α'_1
II_{k-1}
III_k
μ



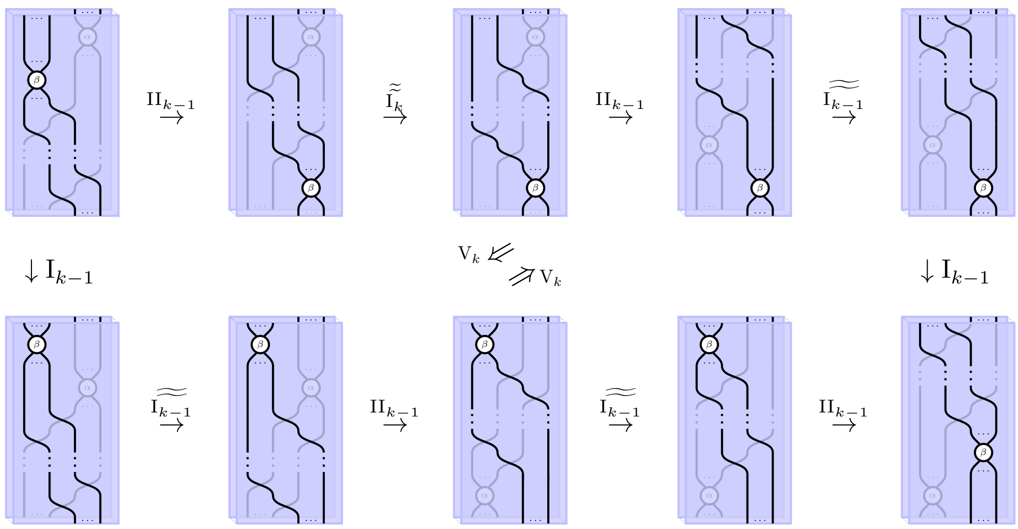
Definition 3.2.13. An $(n+1)$-signature $\sigma = (G_0, ..., G_{n+1})$ supports *interchangers of type* IV_k such that $4 \leq k \leq n$, if for any $(k-1)$-cell $\alpha \in G_{k-1}$ and any $(k-2)$-cells $f_1, ..., f_m \in G_{k-2}$, the following invertible cells are elements of the set G_{k+1}:

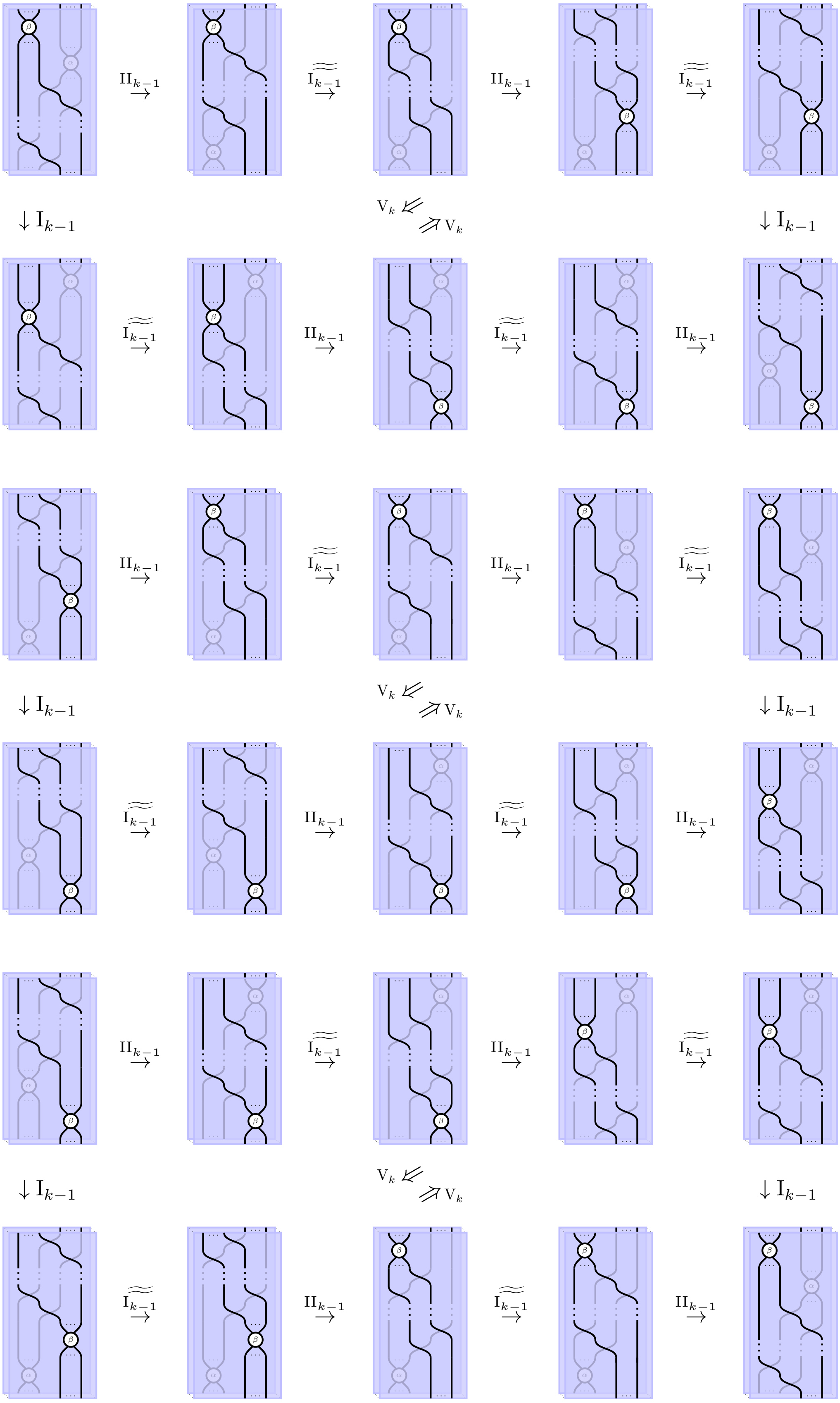


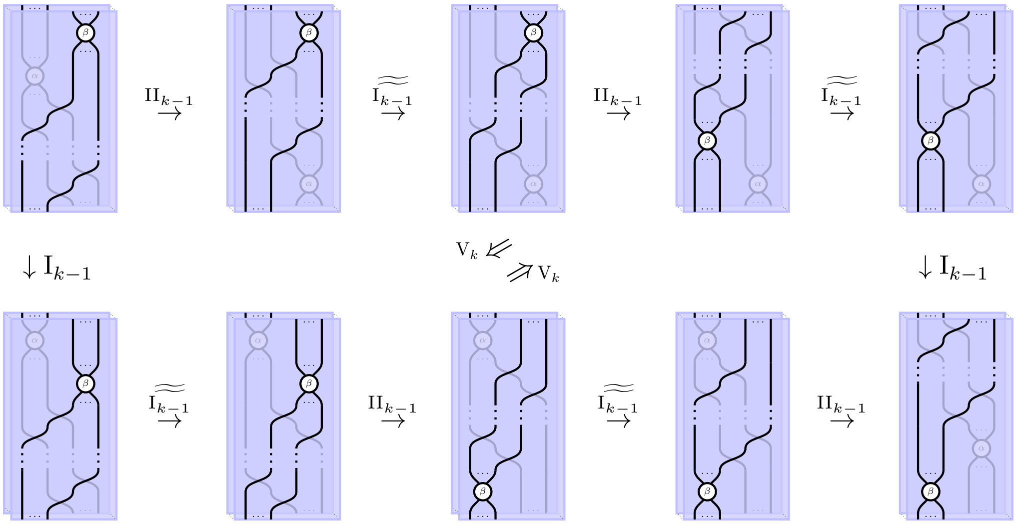

f1 fm
α
II k−1
IV k
I′ k−1

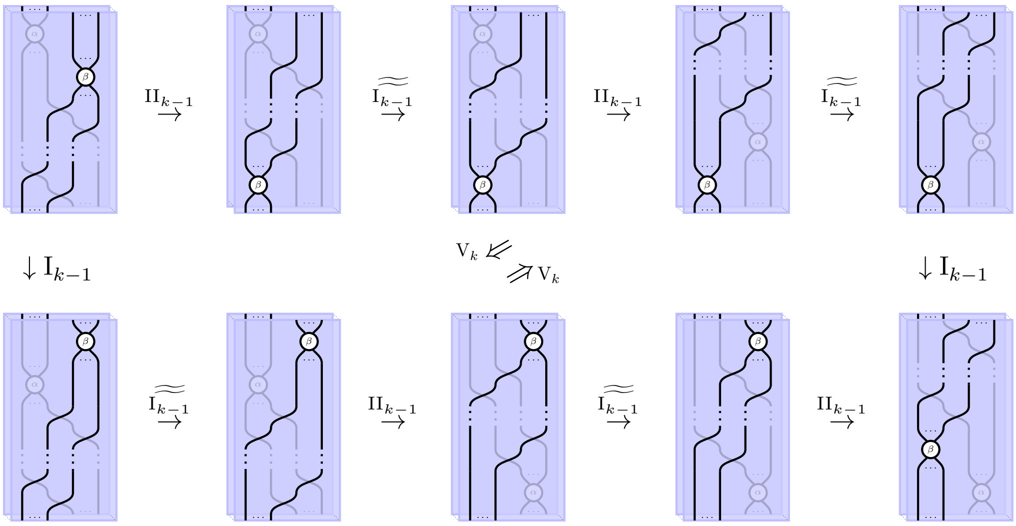


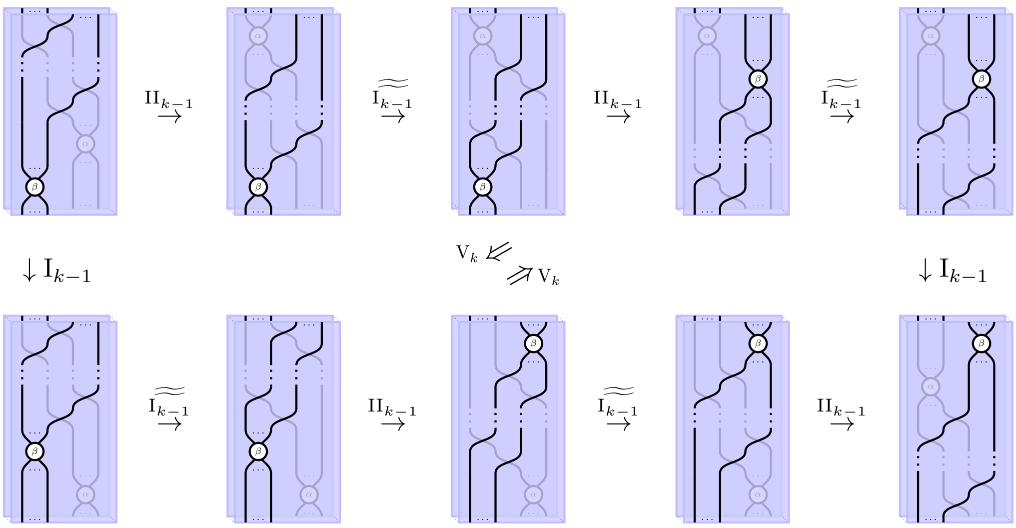
Definition 3.2.14. An $(n+1)$-signature $\sigma = (G_0, ..., G_{n+1})$ supports *interchangers of type* V_k such that $4 \leq k \leq n$, if for any $(k-1)$-cells $\alpha, \beta \in G_{k-1}$, the following invertible cells are elements of the set G_{k+1}:

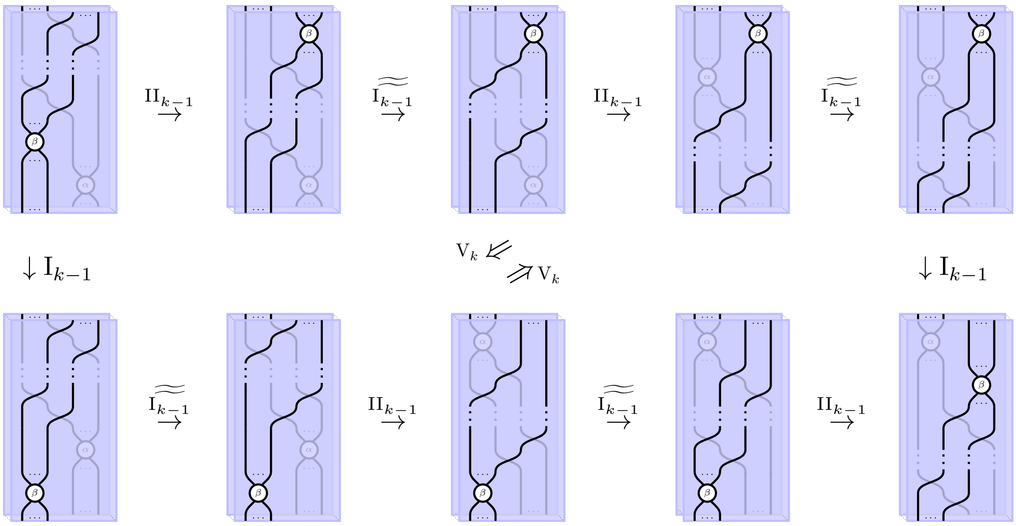



II$_{k-1}$
I$_{k-1}$
V$_k$

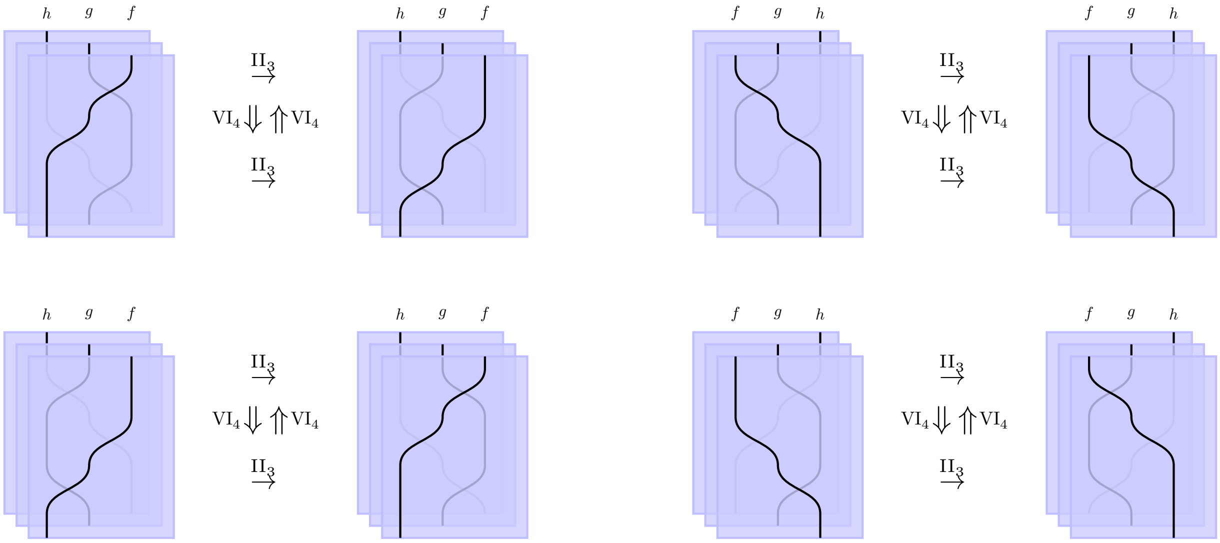

II$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
II$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
↓ I$_{k-1}$
V$_k$
V$_k$
↓ I$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
II$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
II$_{k-1}$


II$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
II$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
↓ I$_{k-1}$
V$_k$
V$_k$
↓ I$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
II$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
II$_{k-1}$


II$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
II$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
↓ I$_{k-1}$
V$_k$
V$_k$
↓ I$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
II$_{k-1}$
$\widetilde{\approx}$ I$_{k-1}$
II$_{k-1}$



Definition 3.2.15. An $(n+1)$-signature $\sigma = (G_0, ..., G_{n+1})$ supports *interchangers of type* VI_k such that $4 \le k \le n$, if for any $(k-2)$-cells $f, g, h \in G_{k-2}$, the following invertible cells are elements of the set G_{k+1}:



3.2.4 Quasistrict n-categories for $n \ge 5$

The approach described here could in principle make it possible to define quasistrict categories of dimensions higher than $n = 5$. Following the pattern of the definitions above, for a quasistrict n-category we would need to take an (n+1)-signature and then list all the necessary singularities that it should support. All the associativity and distributivity axioms are already built into the signature structure and do not have to be listed separately. The challenging part is to enumerate all the n-singularities in an exhaustive fashion and appropriately develop their composite variants. One possible way of doing that is exploring the interactions that different pairs of interchangers have with each other. However this method cannot be considered complete, as exemplified by the Breenator, which cannot be described in this fashion.

Here, we list all the singularities that a quasistrict n-category definitely must contain. We conjecture that, given a quasistrict $(n-1)$-category defined by an n-signature σ, a quasistrict n-category defined by an $(n+1)$ signature σ' must support:

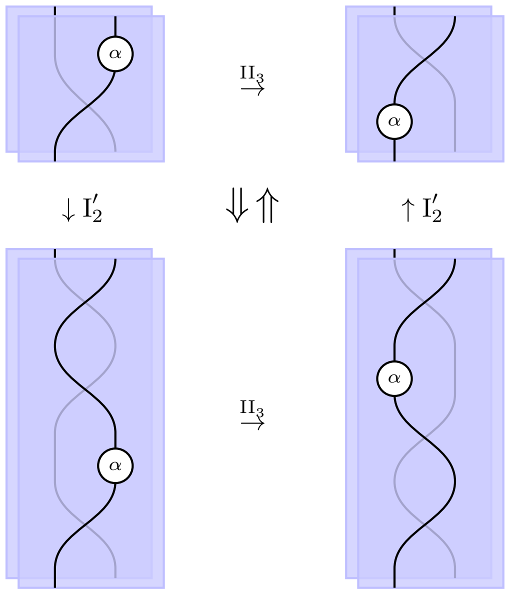
- All singularity types supported by σ.

- All singularity types supported by σ raised by one dimension.
- A singularity type analogous to types II and III where an n-cell is pulled through a 2-cell crossing.

Additionally, there will be higher-level coherences resulting from interactions between interchangers of type III, IV and V which are an interesting topic of further research. The presence of interchangers of type VI in Definition 3.2.3 is evidence that just considering the mutual interactions between $(n-1)$-singularities is insufficient for purposes of an exhaustive approach. Because of this, and all the other reasons outlined above, a general definition for a quasistrict n-category remains elusive.

3.3 A stricter version of quasistrict 3- and 4-categories

Some variants of the interchanger types presented in the previous section are not indispensable to retaining full expressivity, maintaining equivalence to a general weak n-category and keeping the definitions of quasistrict 2-, 3-, 4-categories valid. An instance is presented below:



Here, an application of a variant of type II interchanger is replaced by a combination of applications of another variant of the same interchanger and two applications of the interchanger of type I'. Intuitively, instead of α being pulled down, a tangle in the top part of the diagram is introduced, then α is pulled up and the tangle at the bottom is deleted. However, instead of applying just a single morphism we had to apply multiple morphisms in different sections of the diagram.

A valid approach would be to treat the invertible 5-cells above as the *definition* for this variant of an interchanger of type II_3. In a similar way as for definitions 3.2.6, 3.2.8 and 3.2.10 of composite interchangers, one could then argue that this cell is already present in the signature as a composite of other cells already included there.

Using this method, we could eliminate the need to explicitly add to the signature 4 out of 8 variants of an interchanger of type II_k. All moves that pull a cell through a crossing corresponding to the inverse interchanger of type I_2, as in the example, could be defined in terms of moves that pull the cell through a crossing corresponding to the interchanger of type I_2 and tangling morphisms of type I'_2. This, in turn, would eliminate the need to include different versions of higher level coherences that these 4 variants give rise to. What would remain is 8 out of 16 variants of each of interchanger types III_k and V_k. This would also entirely eliminate the need for an interchanger of type IV. Each of the reduntant variants could be derived only using the axioms for interchangers of type II_3 that involve pulling a cell through the crossing corresponding to an instance of an ordinary interchanger of type I_2 and not the inverse interchanger. This approach would result in a definition of a semistrict 4-category based on the signature structure.

It is true that inclusion of these additional variants introduces more weakness in the definitions. To be entirely consistent with the approach of keeping the definitions as strict as possible, we should refrain from adding these additional variants. However, there is a balance to be kept here. Our overall goal is to maintain equivalence to a general weak n-category and simplify working with the structure being defined by making proofs as short as possible. Strictness of the definitions is just means towards an end, not the overarching aim. For that reason, we introduce the additional variants as, while they introduce some additional weakness, at the same time they make proofs significantly shorter.

The inclusion of these additional interchanger variants is the essence of the difference between the more traditional semistrict approach and quasistrict categories that we propose in this thesis.

3.4 Comparison with 4-teisi

To the best of our knowledge, there has only been one attempt in the literature to define a semistrict 4-category, this is due to Crans [20]. In this section, we discuss this approach and outline how it differs from Definition 3.2.3. We also comment on different treatments of strictness by the two approaches.

In the effort to define a semistrict 4-category, Crans first introduces a tensor product for **Gray**-categories [21], then enriches with respect to the monoidal category built on this tensor product to obtain 4-dimensional categorical structures. As far as terminology is concerned, Crans uses the notion of a *tas* (plural *teisi*), which is Welsh for 'stack' to refer to these. In the same way how a **Gray**-category with identity 3-morphisms is a 2-category, a 4-dimensional tas with only the identity 4-morphisms is a **Gray**-category. In line with this concept, Crans defines a 2-dimensional tas (or a 2D tas for short) to be a 2-category and a 3D tas to be a **Gray**-category.

The key observations are as follows:

- The definition of a 4-tas includes explicit axioms for associativity and distributivity of composition, and for composition with identities. In total, there are 22 variants summarised in 5 clauses. In the approach based on the 5-signature structure, there is no need to explicitly list any of these, as they are built into the structure, as proved for an arbitrary n in Chapter 2.

- Crans only includes 4 variants of interchangers of type II_3, which results in a reducent count for interchangers of types III_4 and V_4 and the omission of IV_4. This treatment of interchangers is equivalent to the more strict approach described in Section 3.3 and results in axiomatic complexity similar to a semistrict 4-category based on the signature structure as discussed in that section.

- Employing the graphical notation allows us to list singularities in a much more systematic fashion, therefore increasing the clarity of the definition, without the need to implicitly refer to reflections and rotations for each axiom.

- The definition of a 4-tas does not account for the Breenator, *i.e.* interchangers of type VI_4, rendering the definition not complete.

The most efficient way of comparing the two approaches is to analyse point by point the individual structures and axioms that Crans proposes and comment on how they relate to the signature structure and interchangers it may support. We cite Crans' 4-tas definition in its entirety and segment it into separate clauses that are directly compared with Definition 3.2.3.

A 4-dimensional tas consists of collections C_0 of objects, C_1 of arrows, C_2 of 2-arrows, C_3 of 3-arrows and C_4 of 4-arrows, together with:

(S1) *Functions $s_n, t_n : C_i \to C_n$ for all $0 \leq n < i \leq 4$ also denoted d_n and $d_n{}^+$ and called n-source and n-target.*

The collection of sets of arrows (cells) and the source and target functions correspond exactly to those in Definition 2.2.1.

(S2) *Functions* $\#_n : C_{n+1} \times_{s_n,t_n} C_{n+1} \to C_{n+1}$ *for all* $0 \leq n < 4$ *called vertical composition.*

This is realised by (vertical) composition of two n-diagrams S, D, as described in Definition 2.4.6.

(S3) *Functions* $\#_n : C_i \times_{s_n,t_n} C_{n+1} \to C_i$ *and* $\#_m : C_{n+1} \times_{s_n,t_n} C_i \to C_i$ *for all* $0 \leq n \leq 2$, $n+1 < i < 4$, *called whiskering.*

This is realised by composition of an n-diagram D and an m-diagram S such that $m \neq n$, as described in Definition 2.4.6.

(S4) *Functions* $\#_n : C_q \times_{s_n,t_n} C_p \to C_{p+q}$ *for all* $0 \leq n \leq 1$, $p, q > n+1$, $p+q-n-1 \leq 4$ *called horizontal composition.*

In our approach this is not a built-in operation, horizontal n-cell composition is instead achieved by first whiskering and then vertical composition.

(S5) *Functions* $\mathrm{id} : C_i \to C_{i+1}$ *for all* $0 \leq n \leq 3$ *called identity.*

This is realised by the identity operation $\mathrm{Id}(D)$ on an n-diagram D, as described in Definition 2.4.22.

Crans also postulates eleven axioms for these structures:

(R1) $\mathbb{C}$ *is a* 4*-truncated globular set.*

We understand that $\mathbb{C} = \{C_0, C_1, C_2, C_3, C_4\}$. The corresponding requirement is realised by the equalities imposed on the source and target maps in Definition 2.2.1.

(R2) *For every* $C, C' \in C_0$ *the collection of elements of* $\mathbb{C}$ *with* 0*-source* C *and* 0*-target* C' *forms a 3D tas* $\mathbb{C}(C, C')$*, with n-composition in* $\mathbb{C}(C, C')$ *given by* $\#_{n+1}$ *and identities given by* id.

Crans requires that for every $C, C' \in C_0$, the collection of elements of $\mathbb{C}$ whose lowest lever source is C and lowest level target C' is a 3D tas correspond to the requirement that σ supports higher-level instances of all singularities supported by a signature presenting a quasistrict 3-category. The difference here is that we list the interchangers explicitly, whereas for Crans they are built into the definition of the lower level structure.

(R3) *For every* $g : C' \to C''$ *in* C_1 *and every* C *and* $C''' \in C_0$, $-\#_0 g$ *is a functor* $\mathbb{C}(C'', C''') \to \mathbb{C}(C', C''')$ *and* $g\#_0-$ *is a functor* $\mathbb{C}(C'', C''') \to \mathbb{C}(C', C'')$.

This is a statement about composition of 1-cells with 0-cells, Crans requires this operation to give rise to functors. In our approach to compose a 0-cell A with 1-cell g, the identity 1-cell on A needs to be produced first. Then, these two are composed by the ordinary composition operation according to Definition 2.4.6. We also explicitly prove that this gives rise to functors in Theorem 2.5.1.

(R4) *For every* $C \in C_0$ *we have:* $s_0(\mathrm{id}_C) = C = t_0(\mathrm{id}_c)$.

This requirement means that sources and targets of a cell that has undergone a transformation by the identity map are equal to the cell itself. In our approach this is a direct consequence of Definition 2.4.1.

(R5) *For every* $C' \in C_0$ *and every* C *and* $C'' \in C_0$, $-\#_0 \mathrm{id}_{C'}$ *is equal to the identity functor* $\mathbb{C}(C', C'') \to \mathbb{C}(C', C'')$ *and* $\mathrm{id}_{C'}\#_0-$ *is equal to the identity functor* $\mathbb{C}(C, C') \to \mathbb{C}(C, C')$.

This is a statement about the relation between the identity operation and 0-composition of 1-cells and that it gives rise to certain functors. In our approach, this is built into the definition of a signature and is captured by Lemma 2.4.24, where we prove that this construction gives rise to functors.

(R6) (a) *For every* $\gamma : f \to f', \delta : g \to g' \in C_2$

$$s_2(\delta\#_0\gamma) = (g'\#_0\gamma)\#_1(\delta\#_0 f)$$
$$t_2(\delta\#_0\gamma) = (\delta\#_0 f')\#_1(g\#_0\gamma)$$

And $\delta \#_0 \gamma$ is an iso-3-arrow.

This interchanger is a 3-cell that corresponds to type I_2, as described in Definition 3.2.5.

(b) *For every $\phi : \gamma \to \gamma' \in C_3$ such that $\gamma, \gamma' : f \to f'$ and $\delta : g \to g' \in C_2$*

$$\begin{aligned} s_3(\delta \#_0 \phi) &= ((\delta \#_0 f')\#_1(g\#_0\phi))\#_2(\delta\#_0\gamma) \\ t_3(\delta \#_0 \phi) &= (\delta\#_0\gamma')\#_2((g'\#_0\phi)\#_1(\delta\#_0 f)) \end{aligned}$$

And $\delta \#_0 \gamma$ is an iso-3-arrow.

This interchanger is a 4-cell that corresponds to type II_3, as described in Definition 3.2.9.

(c) *For every $\gamma : f \to f' \in C_2$ and $\psi : \delta \to \delta' \in C_3$ such that $\delta, \delta' : g \to g'$*

$$\begin{aligned} s_3(\psi \#_0 \gamma) &= (\delta\#_0\gamma')\#_2((g'\#_0\gamma)\#_1(\psi\#_0 f)) \\ t_3(\psi \#_0 \gamma) &= ((\psi \#_0 f')\#_1(g\#_0\gamma))\#_2(\delta\#_0\gamma) \end{aligned}$$

And $\delta \#_0 \gamma$ is an iso-3-arrow.

This interchanger is a 4-cell that corresponds to type II_3, as described in Definition 3.2.9. It is a different variant than in the clause above.

(R7) (a) *For every $\Gamma : \phi \to \phi' \in C_4$ such that $\phi, \phi' : \gamma \to \gamma'$ and $\gamma, \gamma' : f \to f'$ and $\delta : g \to g' \in C_2$:*

$$\begin{aligned} &(((g'\#_0\Gamma)\#_1(\delta\#_0 f))\#_2(\delta\#_0\gamma'))\#_3(\delta\#_0\phi) \\ &\quad = (\delta\#_0\phi')\#_3((\delta\#_0\gamma)\#_2((\delta\#_0 f')\#_1(g\#_0\Gamma))) \end{aligned}$$

This interchanger is a 5-cell that corresponds to type III_4, as described in Definition 3.2.12.

(b) *For every $\gamma : f \to f' \in C_2$ and $\Delta : \psi \to \psi' \in C_4$ such that $\psi, \psi' : \delta \to \delta'$ and $\delta, \delta' : g \to g'$*

$$\begin{aligned} &(\psi'\#_0\gamma)\#_3((\delta'\#_0\gamma)\#_2((g'\#_0\gamma)\#_1(\Delta\#_0 f))) \\ &\quad = (((\Delta\#_0 f')\#_1(g\#_0\gamma))\#_2(\delta\#_0\gamma))\#_3(\psi\#_0\gamma) \end{aligned}$$

This interchanger is a 5-cell that corresponds to type III_4, as described in Definition 3.2.12. It is a different variant than in the clause above.

(c) *For every $\phi : \gamma \to \gamma' \in C_3$ such that $\gamma, \gamma' : f \to f'$ and $\psi : \delta \to \delta' \in C_3$ such that $\delta, \delta' : g \to g'$*

$$\begin{gathered} (((\psi\#_0 f')\#_1(g\#_0\gamma))\#_2(\delta\#_0\phi))\#_3 \\ (((\psi\#_0 f')\#_1(g\#_0\phi))\#_2(\delta\#_0\gamma))\#_3 \\ (((\delta'\#_0 f')\#_1(g\#_0\phi))\#_2(\psi\#_0\gamma))\#_3 \\ = \\ (\psi\#_0\gamma')\#_2((g'\#_0\phi)\#_1(\delta\#_0 f)))\#_3 \\ ((\delta'\#_0\gamma')\#_2((g'\#_0\phi)\#_1(\psi\#_0 f))^{-1})\#_3 \\ ((\delta'\#_0\phi)\#_2((g'\#_0\gamma)\#_1(\psi\#_0 f))) \end{gathered}$$

This interchanger is a 5-cell that corresponds to type V_4, as described in Definition 3.2.14. Depending on the inclusion or exclusion of expansion morphisms in Crans' definition, if the intention was not to include them, but have them to hold as equalities, then reorganisation of crossings, as defined in 3.2.11, is not accounted for here.

There are multiple variants of singularities of types II_3, V_4 that are not present. For type II in Definition 3.2.9 we list eight different variants and for type V in Definition 3.2.14 we list sixteen, which is a direct consequence of the fact that we chose not to make Definition 3.2.3 as strict as possible. In the chosen quasistrict approach, we prioritise simplicity and length of proofs over strictness and, as explained in Section 3.3, we include the additional variants.

(R8) In this clause, Crans discusses expansion morphisms for different singularities. It is not entirely clear whether the intention is to make them actual equalities or higher level morphisms. If the former, then some details pertaining reorganisation of crossings for interchangers of type V_4 are missing from the definition. If the latter, then this would introduce unnecessary weakness into the definition and in violation of the goal of a semistrict definition. As argued before in Section 3.2.1, we do not include explicit expansion morphisms.

(a) *For every* $\gamma : f \to f', \gamma' : f' \to f''$ *and* $\delta : g \to g'$ *in* $\mathbb{C}$*:*

$$\delta \#_0 (\gamma' \#_1 \gamma) = \\ ((\delta \#_0 \gamma') \#_1 (g \#_0 \gamma)) \#_2 ((g' \#_0 \gamma') \#_1 (\delta \#_0 \gamma))$$

This corresponds to a 4-cell decomposing an application of composite interchanger of type I_2 into applications of atomic interchangers. This is not a native structure in our approach. However we could still efficiently reason about decompositions of this type, due to the presence of a shorthand $\widetilde{\mathrm{I}_2}$ as described in Definition 3.2.6. Additionally, this clause corresponds to axiom **S2-17** for a switch 3-category as stated in Definition 3.5.1.

(b) *For every* $\gamma : f \to f'$ *and* $\delta : g \to g', \delta' : g' \to g''$ *in* $\mathbb{C}$*:*

$$(\delta' \#_1 \delta) \#_0 \gamma = ((\delta' \#_0 f') \#_1 (\delta \#_0 \gamma)) \#_2 ((\delta' \#_0 \gamma) \#_1 (\delta \#_0 f))$$

Similar as above, but for the other variant of the interchanger of type I_2.

(c) *For every* $\phi : \gamma \to \gamma', \phi' : \gamma' \to \gamma''$ *such that* $\gamma, \gamma', \gamma'' : f \to f'$ *and* $\delta : g \to g'$ *in* $\mathbb{C}$*:*

$$\delta \#_0 (\gamma'' \#_2 \phi) = \\ ((\delta \#_0 \phi') \#_2 ((g' \#_0 \phi) \#_1 (\delta \#_0 f))) \\ \#_3 \\ (((\delta \#_0 f) \#_1 (g \#_0 \phi')) \#_2 (\delta \#_0 \phi))$$

This is an expansion axiom for interchangers of type II_3. Intuitively, this says that we can pull-through individual subsequent 3-cells through a crossing or pull them through all at once in one move. This is not a built-in structure in our approach, however this does not result in reduced expressivity. Analogously to $\widetilde{\mathrm{I}_2}$ for interchangers of type I, this cell is realised by $\widetilde{\mathrm{II}_3}$ as a composite of applications of atomic interchangers of type II_3, as defined in 3.2.10.

(d) For every $\phi : \gamma \to \gamma'$ such that $\gamma, \gamma' : f \to f'$, for every $\gamma'' : f' \to f''$ and $\delta : g \to g'$ in $\mathbb{C}$:

$$\delta \#_0 (\gamma''' \#_2 \phi) \\ = (((\delta \#_0 \gamma'') \#_1 (g \#_0 \gamma')) \#_2 ((g' \#_0 \gamma'') \#_1 (\delta \#_0 \phi))) \\ \#_3 (((\delta \#_0 \gamma'') \#_1 (g \#_0 \phi)) \#_2 ((g' \#_0 \gamma'') \#_1 (\delta \#_0 \gamma)))$$

This corresponds to an application of a composite interchanger of type II_3, where some additional crossings have to be interchanged out of the way first using interchangers of type I_3. This is not a native cell in our definition, but similarly as above, we could express it as a sequence of atomic interchangers using the shorthand $\widetilde{\mathrm{II}_3}$.

(e) For every $\phi' : \gamma' \to \gamma''$ such that $\gamma', \gamma'' : f' \to f''$, for every $\gamma : f \to f'$ and $\delta : g \to g'$ in $\mathbb{C}$:

$$\delta \#_0 (\phi' \#_2 \gamma) \\ = (((\delta \#_0 \gamma'') \#_1 (g \#_0 \gamma)) \#_2 ((g \#_0 \phi') \#_1 (\delta \#_0 \gamma))) \\ \#_3 (((\delta \#_0 \phi') \#_1 (g \#_0 \gamma)) \#_2 ((g' \#_0 \gamma') \#_1 (\delta \#_0 \gamma)))$$

Similarly as above, but for another variant of the interchanger of type II_3.

(f) For every $\gamma, \gamma' : f \to f'$, for every $\delta : g \to g$; and $\delta' : g' \to g''$ in $\mathbb{C}$:

$$\begin{aligned}
&(\delta' \#_1 \delta) \#_0 \phi \\
&\quad = (((\delta' \#_0 f') \#_1 (\delta' \#_0 \gamma')) \#_2 ((\delta' \#_0 \phi) \#_1 (\delta \#_0 f))) \\
&\quad\quad \#_3 (((\delta' \#_0 f') \#_1 (\delta' \#_0 \phi)) \#_2 ((\delta' \#_0 \gamma') \#_1 (\delta \#_0 f)))
\end{aligned}$$

This is another expansion axiom for interchangers of type II_3. By this, we could either pull a single 3-cell through multiple crossings individually one by one, or we could do this all at once. The same as other expansions above, this is not a native cell in our approach. Again, this does not result in reduced expressivity, as this cell can be realised as a composite of atomic interchangers of type II_3, as defined in 3.2.10.

(g) Crans remarks that there should be additional clauses analogous to the last four, most likely this is to deal with the inverses of the interchanger variants discussed above.

(R9) For every $\gamma : f \to f'$, $\gamma' : f' \to f''$ and for every $\delta : g \to g$;, $\delta' : g' \to g''$ in $\mathbb{C}$:

$$\begin{aligned}
&(\delta' \#_1 \delta) \#_0 (\gamma' \#_1 \gamma) \\
&\quad = ((\delta' \#_0 f'') \#_1 (\delta \#_0 \gamma') \#_1 (g \#_0 \gamma)) \\
&\quad\quad \#_2 ((\delta' \#_0 \gamma') \#_1 (\delta \#_0 \gamma)) \\
&\quad\quad\quad \#_2 ((g'' \#_0 \gamma') \#_1 (\delta' \#_0 \gamma) \#_1 (\delta \#_0 f))
\end{aligned}$$

This is a higher-level coherence which is the result of the expansion axiom for an interchanger of type I_2. There are two different, equivalent methods of decomposition for swapping heights of four adjacent cells and they have to be related to each other by a higher-level morphism. This is a direct evidence for how explicit expansion morphisms give rise to further singularities. The presence of an explicit expansion cell for I_2 suggests that perhaps the intention in the definition was to include higher-level expansion cells explicitly as well.

(R10) *For every* $c \in \mathbb{C}(C, C')_p, c' \in \mathbb{C}(C', C'')_q, c'' \in \mathbb{C}(C'', C''')_r$ *with* $p + q + r \leq 3$ *we have:* $(c'' \#_0 c') \#_0 c = c'' \#_0 (c' \#_0 c)$.

This axiom is on associativity of cell composition. The corresponding result is summarised by equality Eq. (2.21).

(R11) *For every* $c \in \mathbb{C}(C, C')_p, c' \in \mathbb{C}(C', C'')_q$ *such that* $p, q > 0$ *and* $p + q \leq 3$ *if* $q \leq 2$ *we have:* $c' \#_0 \mathrm{id}_c = \mathrm{id}_{c' \#_0 c}$ *and if* $p \leq 2$ *we have:* $c' \#_0 \mathrm{id}_c = \mathrm{id}_{c' \#_0 c}$.

This axiom defines how to compose a cell with an identity of a cell of a lower dimension. The corresponding result is proved in Lemma 2.4.25.

Each clause in Crans' definition is concluded with comments that each of these cells should also be an isomorphism, as this ensures that their inverses are also included as cells and decreases the scope of singularities that are not included. This corresponds to inclusion of four variants of singularities of type II_3, as described in Definition 3.2.13.

To summarise, the definition proposed by Crans has a different approach to strictness than the definition of a quasistrict 4-category. There are two singularity types that are not present in this definition, namely: IV_4 and VI_4. The omission of the former is the intended consequence of adapting an approach stricter than in Definition 3.2.3. Singularity of type VI_4 should have been included for the definition to be correct.

3.5 Satisfaction of switch 3-category axioms

In this section, we discuss the definition of a switch 3-category given by Douglas and Henriques [22], which is an alternative presentation of a **Gray**-category. We concentrate on showing how it corresponds to Definition 3.2.2 of a quasistrict 3-category as a 4-signature. Since our definition of a quasistrict 4-category is a natural extension of the quasistrict 3-category definition, we consider satisfaction of switch 3-category axioms as strong evidence for correctness of this new, proposed definition.

Definition 3.5.1. *Switch 3-category*: See Appendix B.

The summary of the definition is as follows:

- The definition lists:
  - ⋆ Four sets of data: 0-cells, 1-cells, 2-cells and 3-cells all satisfying globularity conditions.
  - ⋆ Thirteen maps between these sets of cells: identities, composition and whiskering maps.
  - ⋆ Thirty four axioms that these maps are supposed to satisfy.
- Maps have the following properties:
  - ⋆ The identity map applied to a k-cell f produces a $(k+1)$-cell whose both source and target are f.
  - ⋆ Composition of two k-cells is the usual categorical notion of $(k-1)$-composition or vertical composition.
  - ⋆ Whiskering allows us to avoid ambiguity when composing two k-cells. Two k-cells α and β can only be composed vertically, but whiskering using cells of lower dimensions stil enables construction of any arbitrary k-cell.

  Thirty four axioms regulate how these maps interact with each other, the list consist of various associativity and distributivity results.

There are four axioms for singularities of type I_2 (**S2-16 - S2-19**) plus one axiom each for singularities of types I_3 (**S2-25**) and II_3 (**S2-30**). The 'switch' map, which lends the name to the entire category, corresponds to interchangers of type I_2. The most significant difference between our approach and the definition by Douglas and Henriques is the inclusion of the expansion axiom for for the 'switch' map (**S2-17**). As argued in Section 3.2.1, an expansion morphism for I_2 is not a native structure in Definition 3.2.2, however this does not prevent us from retaining full expressivity. This is because, a composite interchanger of type I_2 can be expressed as a series of atomic interchangers of the same type using the shorthand $\widetilde{\mathrm{I}_2}$, as described in Definition 3.2.6. The final remark is that, Douglas and Henriques ascertain existence of inverses for the switch morphism, which corresponds to inclusion of higher invertibility cells of type $\widetilde{\mathrm{I}_2}$.

Overall, the bulk of axioms listed in the definition of a switch 3-category pertains associativity and distributivity of composition, as well as composition with identity. In Definition 3.2.2 of a quasistrict 3-category based on the 4-signature structure, these do not have to be listed explicitly, as they are already implicitly built into the structure, as proved for an arbitrary n in Chapter 2. This results in a much lower axiomatic count of the quasistrict 3-category definition. The results on associativity and distributivity of composition proved in Chapter 2 are of non-trivial complexity which for $n = 3$ perhaps does not outweight manually listng all the axioms in the definition. However, the main advantage of the approach based on the signature structure is that these results are proved for an arbitrary n and therefore could be used for $n > 3$ with no additional effort. This is exemplified by the comparison with the 4-tas definition in Section 3.4.

Theorem 3.5.2. *Definition 3.2.2 satisfies the axioms for a switch 3-category listed by Definition 3.5.1.*

Proof. Given a switch 3-category defined in accordance with Definition 3.5.1, consider a 4-signature $\sigma = \{T_0, T_1, T_2, T_3, T_4\}$ that supports interchangers of types I_2, I_3 and II_3. Since σ is a signature, there are maps $s_k, t_k : T_k \to T_{k-1}$ for $k = \{1, 2, 3\}$ such that $s_k \circ t_k = s_k \circ s_k$ and $t_k \circ t_k = t_k \circ s_k$, as required.

Now let us consider individual maps using the naming scheme that Douglas and Henriques use:

1-data

- **S1-1** For a 0-diagram D, the map i_x is realised by: $i_x(D) := \mathrm{Id}(D)$
- **S1-2** For two 1-diagrams S, D the map m_x is realised by $m_x(S, D) := S \circ D$
- **S1-3** For a 1-diagram D, the map i_x is realised by: $i_x(D) := \mathrm{Id}(D)$
- **S1-4** For two 2-diagrams S, D the map m_y is realised by $m_y(S, D) := S \circ D$
- **S1-5** For a 1-diagram D and a 2-diagram S, the map w_r is realised by $w_r(S, D) := S \circ D$
- **S1-6** For a 2-diagram D and a 1-diagram S, the map w_l is realised by $w_l(S, D) := S \circ D$
- **S1-7** This is realised by interchangers of type I_2 supported by σ.
- **S1-8** For a 2-diagram D, the map i_z is realised by: $i_z(D) := \mathrm{Id}(D)$
- **S1-9** For two 3-diagrams S, D the map m_z is realised by $m_z(S, D) := S \circ D$
- **S1-10** For a 3-diagram D and a 2-diagram S, the map f_b is realised by $f_b(S, D) := S \circ D$
- **S1-11** For a 2-diagram D and a 3-diagram S, the map f_t is realised by $f_t(S, D) := S \circ D$
- **S1-12** For a 3-diagram D and a 1-diagram S, the map h_r is realised by $h_r(S, D) := S \circ D$
- **S1-13** For a 1-diagram D and a 3-diagram S, the map h_l is realised by $h_l(S, D) := S \circ D$

1-morphism axioms

- **S2-1** Given that map m_x is realised by diagram composition and map i_x by the identity operation, we need to show that $\mathrm{Id}(S) \circ D = D$ for any 1- diagram D and any 0-diagram S such that $\mathrm{Id}(S) \circ D$ exists. This follows by Lemma 2.4.24.
- **S2-2** The argument is analogous to **S2-1**.
- **S2-3** Given that map m_x is realised by diagram composition, we need to show that $S \circ (D \circ M) = (S \circ D) \circ M$ for any three 1-diagrams S, D, M that are composable. This follows by equality Eq. (2.21).
- **S2-4** Given that map m_y is realised by diagram composition and map i_y by the identity operation, we need to show that $\mathrm{Id}(S) \circ D = D$ for any 2- diagram D and any 1-diagram S such that $\mathrm{Id}(S) \circ D$ exists. This follows by Lemma 2.4.24.
- **S2-5** The argument is analogous to **S2-4**.
- **S2-6** Given that m_y is realised by diagram composition, we need to show that $S \circ (D \circ M) = (S \circ D) \circ M$ for any three 2-diagrams S, D, M that are composable. This follows by equality Eq. (2.21).
- **S2-7** Given that map w_r is realised by diagram composition, we need to show that for any two 1-diagrams S, D that are composable, we have: $\mathrm{Id}(S) \circ D = \mathrm{Id}(S \circ D)$. This follows by Lemma 2.4.25.
- **S2-8** The argument is analogous to **S2-7**.
- **S2-9** Given that maps m_y and w_r are realised by diagram composition, we need to show that $(S \circ D) \circ M = (S \circ M) \circ (D \circ M)$ for any two 2-diagrams S, D and a 1-diagram M that are composable. This follows by equality Eq. (2.23).
- **S2-10** The argument is analogous to **S2-9** and the result follows by equality Eq. (2.22).
- **S2-11** Given that maps m_x and w_l are realised by diagram composition, we need to show that $(S \circ D) \circ M = S \circ (D \circ M)$ for any two 1-diagrams D, M and a 2-diagram S that are composable. This follows by Theorem 2.21.

- **S2-12** The argument is analogous to **S2-11**.
- **S2-13** Given that maps w_l and w_r are realised by diagram composition, we need to show that $(S \circ D) \circ M = S \circ (D \circ M)$ for any two 1-diagrams S, M and a 2-diagram D that are composable. This follows by Theorem 2.21.
- **S2-14** Given that map m_y is realised by diagram composition and map i_x by the identity operation, we need to show that $\mathrm{Id}(S) \circ D = D$ for any 2- diagram D and any 0-diagram S such that $\mathrm{Id}(S) \circ D$ exists. This follows by Lemma 2.4.24.
- **S2-15** The argument is analogous to **S2-14**.
- **S2-16** Given that the 'switch' map corresponds to interchangers of type I_2, this holds by Definition 3.2.5. As application of an interchanger of type I at height $h = 1$ for $|D| = 1$ has no effect on D.
- **S2-17** This is an expansion axiom for the 'switch' morphism, it is not present in Definition 3.2.2. For reasons outlined earlier in this section, an explicit expansion morphism for composite interchangers does not add any additional expressivity. In σ, a composite interchanger of type I_2 can be expressed as a sequence of atomic interchangers of the same type using the construction $\widetilde{\mathrm{I}_2}$ described in Definition 3.2.6.
- **S2-18** This follows by the result on associativity of composition 2.21 and by the fact that σ supports interchangers of type I_2. Since we only interpret the interchanger in the context of the digram D at the particular height h, the order in which any additional 1-cells on either side are composed in does not alter the overall interchange, as required.
- **S2-19** The argument is analogous to **S2-18**, but for 1-cells that are in between the 2-cells being interchnaged. By Definition 3.2.5, there may be an arbitrary number of them and the order in which they are composed with 2-cells has no effect on the interchange, as required.
- **S2-20** Given that map m_z is realised by diagram composition and map i_z by the identity operation, we need to show that $S \circ \mathrm{Id}(D) = S$ for any 3- diagram S and any 2-diagram D such that $S \circ \mathrm{Id}(D)$ exists. This follows by Lemma 2.4.24.
- **S2-21** Given that map m_z is realised by diagram composition, we need to show that $S \circ (D \circ M) = (S \circ D) \circ M$ for any three 3-diagrams S, D, M that are composable. This follows by equality Eq. (2.21).
- **S2-22** Given that map f_t is realised by diagram composition, we need to show that for any two 2-diagrams S, D that are composable, we have: $\mathrm{Id}(S) \circ D = \mathrm{Id}(S \circ D)$. This follows by Lemma 2.4.25.
- **S2-23** Given that map f_t is realised by diagram composition and map i_y by the identity operation, we need to show that $S \circ \mathrm{Id}(D) = S$ for any 3- diagram S and any 1-diagram D such that $S \circ \mathrm{Id}(D)$ exists. This follows by Lemma 2.4.24.
- **S2-24** Given that maps m_z and f_t are realised by diagram composition, we need to show that $(S \circ D) \circ M = (S \circ M) \circ (D \circ M)$ for any two 3-diagrams S, D and a 2-diagram M that are composable. This follows by equality Eq. (2.23).
- **S2-25** This is an instance of two 3-cells being subject to the interchange law, similarly as **S1-7** for 2-cells. This axiom holds, since σ supports interchangers of type I_3.
- **S2-26** Given that maps m_y and f_t are realised by diagram composition, we need to show that $S \circ (D \circ M) = (S \circ D) \circ M$ for any two 2-diagrams D, M and a 3-diagram S that are composable. This follows by equality Eq. (2.21).

- **S2-27** Given that maps f_t and f_b are realised by diagram composition, we need to show that $(S \circ D) \circ M = S \circ (D \circ M)$ for any two 2-diagrams S, M and a 3-diagram D that are composable. This follows by equality Eq. (2.21).

- **S2-28** Given that map h_r is realised by diagram composition, we need to show that for a 2-diagram S, and a 1-diagram D that are composable, we have: $\mathrm{Id}(S) \circ D = \mathrm{Id}(S \circ D)$. This follows by Lemma 2.4.25.

- **S2-29** Given that map h_r is realised by diagram composition and map i_x by the identity operation, we need to show that $S \circ \mathrm{Id}(D) = S$ for any 3- diagram S and any 0-diagram D such that $S \circ \mathrm{Id}(D)$ exists. This follows by Lemma 2.4.24.

- **S2-30** This is an instance of naturality of the switch morphism (**S1-7**) in one of its inputs and corresponds to interchangers of type II. This axiom holds, since σ supports interchangers of type II_3.

- **S2-31** Given that maps m_z and h_r are realised by diagram composition, we need to show that $(S \circ D) \circ M = (S \circ M) \circ (D \circ M)$ for any two 3-diagrams S, D and a 1-diagram M that are composable. This follows by equality Eq. (2.23).

- **S2-32** Given that maps f_t and h_r are realised by diagram composition, we need to show that $(S \circ D) \circ M = (S \circ M) \circ (D \circ M)$ for a 3-diagrams S, a 2-diagram D and a 1-diagram M that are composable. This follows by equality Eq. (2.23).

- **S2-33** Given that maps m_x and h_r are realised by diagram composition, we need to show that $S \circ (D \circ M) = (S \circ D) \circ M$ for any two 1-diagrams D, M and a 3-diagram S that are composable. This follows by Theorem 2.21.

- **S2-34** Given that maps h_l and h_r are realised by diagram composition, we need to show that $(S \circ D) \circ M = S \circ (D \circ M)$ for any two 1-diagrams S, M and a 3-diagram D that are composable. This follows by Theorem 2.21.

All additional axis flips required by Definition [22] follow in an analogous way. ☐

3.6 Further results

To conclude this chapter, we discuss how signatures supporting the appropriate types of interchangers reproduce certain standard categorical structures. Recall a segment of the periodic table for weak higher categories presented in Section 1.2.3:

	2	3	4
0	2Cat	3Cat	4Cat
1	MonCat	Mon2Cat	Mon3Cat
2	CommMon	BrMonCat	BrMon2Cat
3	-	CommMon	SymMonCat

We build up to prove that a degenerate quasistrict 4-category with the bottom three levels trivialised is a symmetric monoidal category, as predicted by the table above. First we show how to recreate a strict 2-category using the signature structure:

Theorem 3.6.1. *A 3-signature σ supporting interchangers of type I_2 is the same as a strict 2-category.*

Proof. Interchangers of type I_2 act on 2-cells and they are 3-cells themselves. But in a 3-signature, all 3-cells are identities, so by this, we obtain strict interchange laws and hence a strict 2-category. ☐

As a direct consequence of the results in the previous sections, we also obtain a **Gray**-category:

Theorem 3.6.2. *A 4-signature σ supporting interchangers of types* I_2*,* I_3 *and* II_3 *is the same as a* ***Gray****-category.*

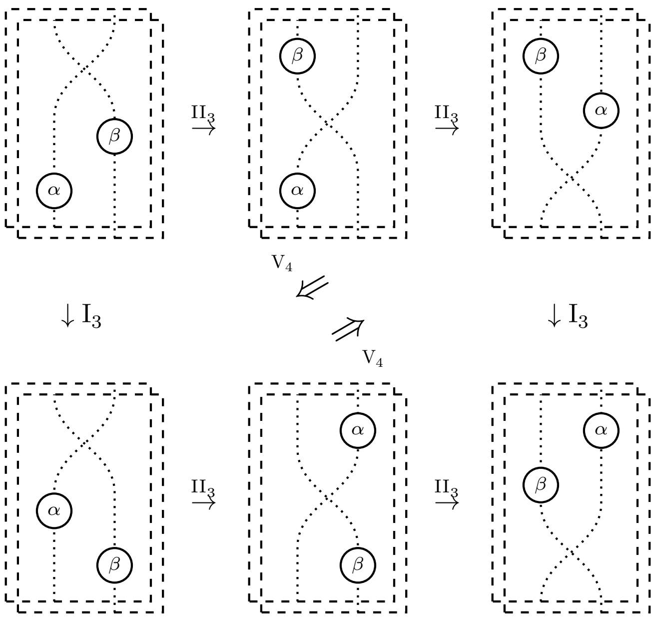
Proof. This follows, since by Theorem 3.5.2 σ satisfies axioms of a switch 3-category 3.5.1, which is a an alternative presentation of a **Gray**-category. □

As a result, we argue that a degenerate instance of a 5-signature is a symmetric monoidal category.

Theorem 3.6.3. *A 5-signature σ supporting interchangers of types* I_2, I_3, II_3, I_4, II_4, III_4, IV_4, V_4 *and* VI_4 *that has one 0-cell, one 1-cell and one 2-cell is the same as a symmetric monoidal category*

Proof. Using a standard coherence result, a **Gray**-category is equivalent a weak 3-category. Then, by the periodic table for higher categories we obtain that a **Gray**-category with one 0-cell and one 1-cell is a braided monoidal category. Hence, since σ supports interchangers of types I_3, I_4 and II_4, by Theorem 3.6.2 we get that σ is at least a braided monoidal category.

What remains to be shown is that the braiding, which is given by interchangers of type I_3, is also a symmetry. This is achieved by interchangers of type V_4. Since the bottom three levels of σ are trivialised, in the graphical representation we could draw them as empty spaces, empty sheets and empty lines. That way in each variant of V_4, both involved morphisms α and β are scalars moving through the empty space and exchanging heights. The graphical illustration is as follows:

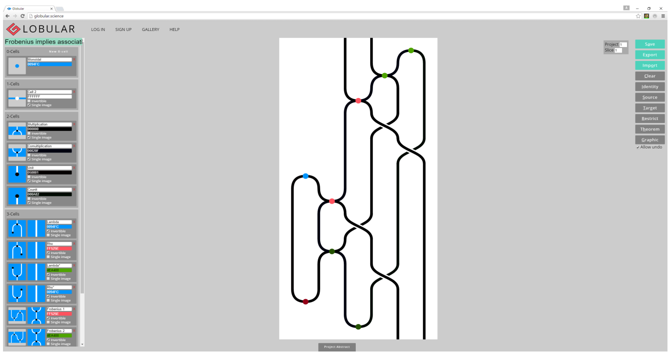


The dotted and dashed lines are not a part of the graphical representation, they are included to give a sense of where the trivialised cells are. Note that V_4 switches between the two different variants of the interchanger of type I_3 acting on α and β . In one method α passes β on the *right* in the other, on the *left.* This way we obtain an isomorphism between the two braidings and hence, a symmetric monoidal category, as required. □

Chapter 4

Automated rewriting for quasistrict higher categories: Globular

Globular is a proof assistant for formalisation and verification of higher categorical proofs based on the theory of higher dimensional quasistrict categories presented in Chapters 2 and 3. In this chapter, we first discuss the technologies used to implement the system and describe its key functionalities. We then proceed to present the main algorithms and explain their relationships with the diagram modifying operations of rewriting and composition. We conclude the chapter by illustrating the functionalities of Globular with a fully-worked example, where a signature used for the proof of Theorem 1.3.4 is built from scratch. Finally, we present a list of other results that have been formalised in Globular by its community of users. This is accompanied by direct links to the gallery of public projects on the Globular webpage. An example project looks as follows:



As hinted at in Chapter 1, Globular adapts the perspective of higher dimensional rewriting, wherein a multi-step proof is viewed as a sequence of individual k-cell rewrites between lower level objects. Additionally, the notion of equality between k-cells f, g is captured by an invertible $(k+1)$-cell $f \xrightarrow{\alpha} g$. Here, α witnesses that f can be rewritten into g and α^{-1} witnesses the converse rewrite. In particular, if f is a part of some composite $C[f]$, the invertible cell α also witnesses the equality $C[f] = C[g]$. This is in accordance with the basic premise of homotopy type theory, where proofs can themselves be subject to higher-level relations.

Globular produces graphical visualisations of higher-dimensional proofs based on the graphical formalism for diagram structures as described in Definition 3.1.1. As discussed in Chapter 2, the set of all k-diagrams over a signature σ can be thought of as the set of all composite k-cells generated using the cells in σ. Since in the perspective of higher dimensional rewriting every higher categorical proof is also a cell, it can be visualised as a diagram as described in Definition 2.2.2.

The two primitive structures that Globular operates on are signatures and diagrams, as defined in Chapter 2. Their mutually-recursive nature allows the user to build a signature, which can be thought of as the set of generating elements for a category, in parallel with building increasingly sophisticated diagram structures. The tool makes it possible to rewrite and compose diagrams, and then make them sources or targets of higher-level cells to be added to the signature. When both the source and the target n-diagram of a potential $(n+1)$-cell are designated, Globular conducts a type check for satisfaction of globularity conditions and automatically prevents improper cells from being formed.

The tool has been designed with user experience in mind, so that it could be a proof assistant that is broadly used in the category theory community. For this reason, Globular is hosted on the web at `http://globular.science` and can be used directly in the browser, without requiring the user to download and install a piece of software on their machine. We believe that the intuitive user interface, the low barrier to entry and the ability to hyperlink proofs directly into research papers will make Globular a popular tool among the higher category theory community.

At present, Globular is capable of modelling quasistrict n-categories, as defined in Chapter 3, for $n \leq 4$. Additionally, it supports singularities of types $\mathrm{I} - \mathrm{VI}$ for an arbitrary $n \geq 5$. As new types of singularities for higher levels are explored and formally classified, they can be added to the catalogue in Globular and enhance the capabilities of the tool. There is no other tool available in the community that would have similar abilities.

The closest comparable attempts at automated reasoning for category theory are: Quantomatic [32], Opetopic [24], and a tool that could be built basing on Mimram's work on n-polygraphs [42] as presentations of strict n-categories. All these tools are quite different in nature than Globular. Quantomatic is a set of tools for automated rewriting of string graphs and implements the theory of symmetric monoidal categories. In that, it models only a subset of categories that could be modelled in Globular. On the other hand, its capabilities with regards to automatically finding rewrite paths and synthesising theories are far superior. Opetopic is based on the concept of modelling weak n-categories with opetopic shapes, which are best described as higher dimensional analogues of the tree structure. This approach offers greater expressivity, in fact going as high as an arbitrary n, however it disturbs the crucial topological aspect of the proofs. Therefore, despite being a graphical notation, it is very different from ours and offers limited insight into the inner structure of cells being modelled. Additionally, Opetopic is less well-developed with regards to user interface features, for instance lacking the capability to save and retrieve proofs. Finally, higher dimensional rewriting systems based directly on the definition of an n-polygraph only model strict n-categories, and are therefore less expressive than the approach taken in Globular.

The contents of this Chapter are based on a joint paper with Vicary and Kissinger [7], included is only the material that this author contributed significantly towards.

4.1 System description

Globular is implemented in Javascript and runs client-side embedded in the web browser, with all the computation taking place on the user's machine, therefore limiting the need for data transfer. The back-end is a Node.js server responding to user's requests and hosting an account system for users that allows them to register and privately save working versions of proofs and then continue the work on different machines. However, to lower the barrier to entry, there is no requirement to register for an account to be able to use the tool. The functionalities of exporting and importing a project can be used to directly generate JavaScript Object Notation (JSON) files that can be saved on the user's machine. When the user is satisfied with the finished proof they can make it public and share it with the rest of community, the proof is then added to Globular gallery and a unique URL linking to the proof generated. We give examples of this functionality at the end of this chapter. The entire project is open-source, and the code is available at `globular.science/source`.

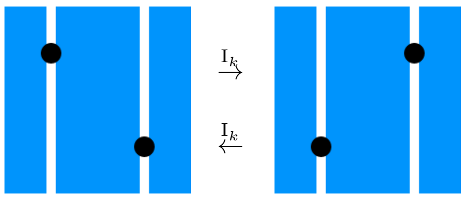
While the weak formal structure of Javascript is a disadvantage, formal verification of the code is not a priority for us. We acknowledge that implementing a fully typed structure would make the implementation less susceptible to errors, but it would not remove the need to give rigorous proofs of the correctness of implementation. This has however never been the focus and we instead concentrate on proving correctness of the theoretical model. At present, the main priority for Globular is to be an intuitive proof assistant that is widely used by the community. But, perhaps in the future, it would be interesting to formalise the approach to typing, as the theoretical foundation for that is sound and already available.

The interface has been designed to be friendly and intuitive. Diagrams can be created, rewritten and composed by clicking elements in the signature and selecting an attachment point from the list of options. Given an n-diagram D in the workspace, the operation triggered depends on the dimension k of the cell $g \in G_k$ that we select from the signature σ. If $k = n + 1$, then D gets rewritten; if $k < n + 1$, D is subject to composition. For the latter, first implicitly a diagram $S = i(g)$ of the generator g is created, this corresponds to the embedding $i : G_k \to {\Delta^*}_k$, as described in Definition 2.2.4. Diagrams S and D then get composed in accordance with Definition 2.4.6 and the algorithm described in the next section. If $k > n + 1$ there is no effect on D and the tool asks the user to select another cell.

However, selecting elements from the signature is not the only method of modifying the diagram in the signature. Interchanger morphisms of types I-VI can be applied directly by clicking and dragging the appropriate cells within the diagram.

The graphical visualisations of cells are generated using SVG, however this limits the rendering to a maximum of two dimensions. This may be regarded as a serious difficulty, especially when dealing with higher dimensional structures. For that reason, in the future, we intend to implement a 3D graphics engine using Three.js. However, even these enhanced graphical capabilities will not be sufficient to work efficiently with structures of dimension $n = 4$ and higher. To work around that, we implemented a system of toggles, that allows to suppress the lowest dimensions and view slices that are of interest. Even though, at times, this may prove cumbersome, it is certainly worthwhile as this solution provides us with a systematic method for viewing morphisms in any n-dimensional structure. For an n-diagram D such that $n \geq 3$, for which the number of dimensions projected out is k, there are $n - k - 2$ slice toggles that allow us to view a multidimensional structure as a sequence of 2D slices.

- For interchangers of type I_k, as described in Definition 3.2.5, the vertex to be interchanged needs to be clicked and dragged into the direction of the intended swap to obtain the following graphical effect:

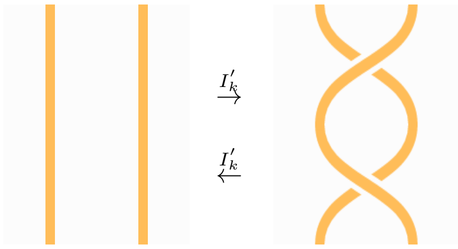
  

  In the view, where the bottom dimension has been projected out, the two variants of the interchanger of type I_k look as follows:

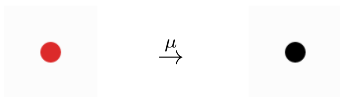
  Note that the familiar image of the braiding is retrieved. This image is similar to the custom-drawn graphics in Chapter 3, the difference is that the view of the rear sheets via transparency is missing. Here, it is really as if we were looking at the 3D graphical representation of a 3-cell 'side on'.

- Inverses for all interchanger types are handled in a uniform fashion, recall the notation f' to indicate a higher level invertibility cell, as described in Definition 3.2.7. For type I_k, to introduce an instance of an interchanger followed by its inverse, one of the wires needs to be clicked and dragged left or right to tangle the wires:



To cancel an interchanger and its adjacent inverse, simply drag one of them into the direction of the other. Both of these can be depicted in the projected view with the bottom dimension projected out:

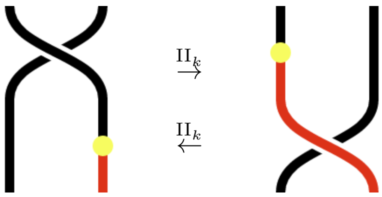
- For interchangers of type II_k, as described in Definition 3.2.9, consider the following k-cell μ:



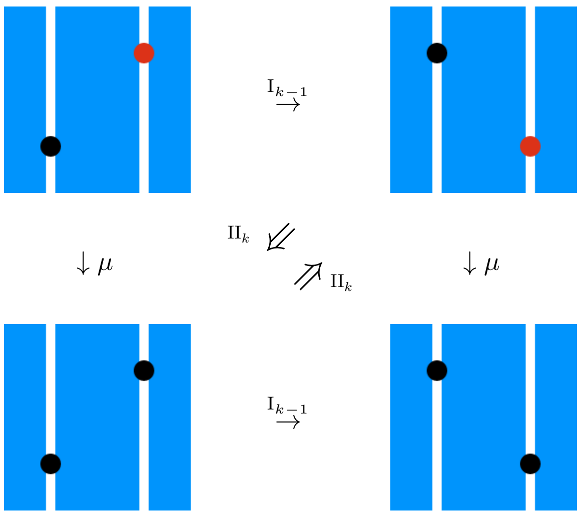
In the view, with the bottom two dimensions projected out, the cell looks as follows:

μ := 

To trigger an interchanger of type II_k, a vertex representing a k-cell needs to be clicked and pulled through an adjacent crossing:



Unprojecting one dimension, we can view a sequence of slices for both the source and the target of this interchanger:



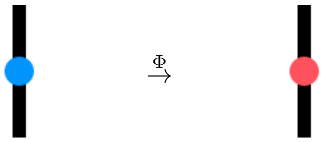
While in a view with the three bottom dimensions projected out, an interchanger of type II_k is represented in a single picture:

$\mathrm{II}_3 \quad =$ 

Notice how the path traced out by μ (coloured yellow) crosses to the other side of the path representing the interchanger cell I_{k-1}.

For interchangers of types III_k, IV_k and V_k, we need to project out two dimensions to view their source and target as a sequence of slices. To view each interchanger as a single cell, we need to project out the three bottom dimensions.

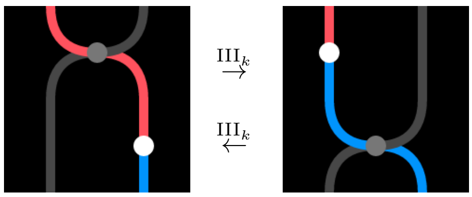
- For type III_k, as described in Definition 3.2.12, consider the following k-cell Φ:



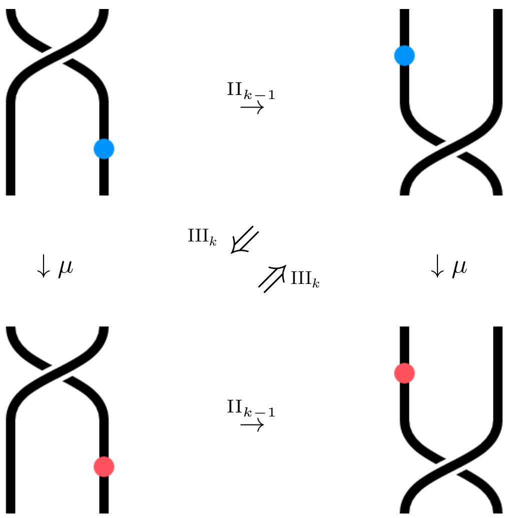
In the projected view it looks as follows:

$\Phi \quad :=$ 

Now, a vertex representing a k-cell Φ needs to be clicked and pulled through an adjacent vertex representing an interchanger of type II_{k-1}:

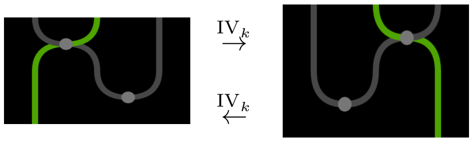


In the unprojected view, where only two bottom dimensions have been suppressed:

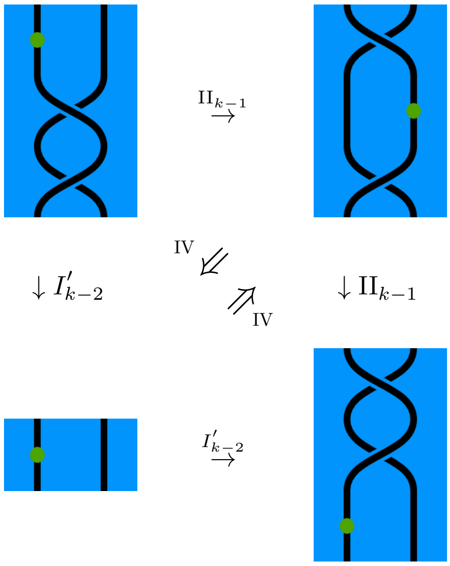


- For type IV_k, as described in Definition 3.2.13, a cell representing an interchanger of type II_{k-1} needs to be clicked and dragged in the direction of a collection of adjacent vertices representing a

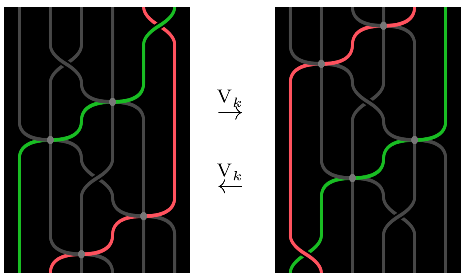
$(k-1)$-cell of type I$'$ that introduces two consecutive inverse applications of an interchanger of type I$_{k-2}$:



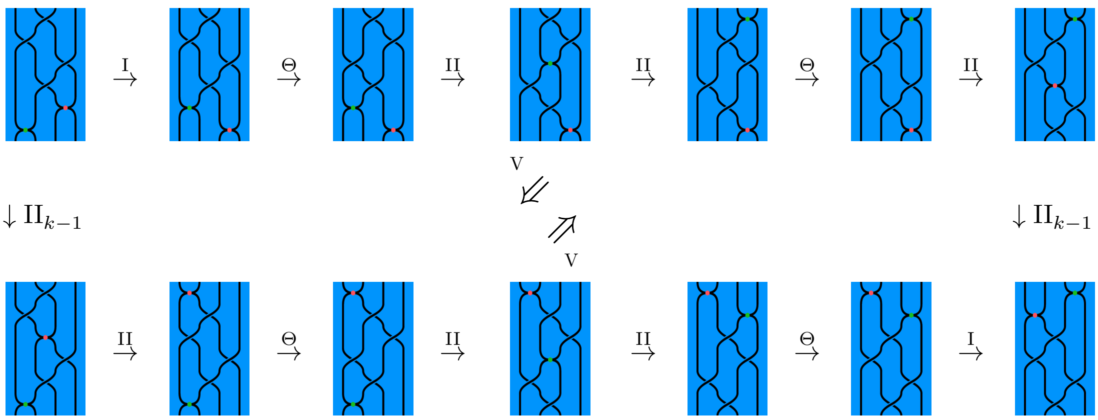
In the unprojected view, with the bottom two dimensions suppressed:



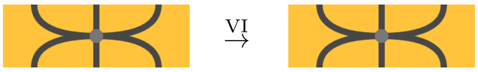
- For type V$_k$, as described in Definition 3.2.14, a cell representing an interchanger of type I$_{k-1}$ and depicted as a crossing of red and green wires needs to be clicked and pulled through an adjacent collection of vertices representing interchangers of type II$_{k-1}$ and crossing reorganisations Θ, as defined i in Definition 3.2.11:



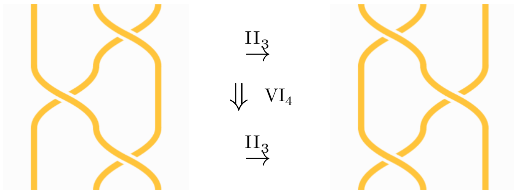
In the unprojected view, with the bottom two dimensions suppressed:



- For type VI_k, as described in Definition 3.2.15, a 4-cell representing an interchanger of type II_{k-1} applied to an instance of an interchnager of type I_{k-2} needs to be clicked to transform it into a different interchanger of the same type. Graphically this does not have any effect when viewed in the projected perspective:

  Similarly in the unprojected view, the graphical representations look the same. However in the first instance, the interchanger of type II_{k-1} is applied at height 0 and the bottom right crossing is pulled up, in the second instance II_{k-1} is applied at height 2 and the top right crossing is pulled down.

For interchangers of types II_k, III_k, IV_k and V_k a pattern starts to emerge, in which application of an interchanger is graphically visualised as pulling one cell through another. This is not surprising, since all these interchangers can be interpreted to capture some naturality condition. An important note to be made is that all these interchangers get executed only if they indeed are valid moves for that segment of the diagram, which is checked by the Globular engine.

There are several additional features in Globular that are intended to make navigation of multi-dimensional diagrams more intuitive:

- A name and colour can be chosen for every cell. Once a colour gets changed for a cell g in the signature, then it is updated for all higher-level cells that g is a component of.
- For a diagram in the workspace, the names of its individual cells $D[i].g$ get displayed in the form of pop-up labels, when a cell is hovered over. For interchangers, pre-assigned names for their different variants get displayed.
- In addition to the name of a cell $D[i].g$, a list of its coordinates within the diagram gets displayed. This corresponds to the data contained in the embedding $D[i].e$, which embeds the source of g in the appropriate slice of the diagram.
- Invertibility of cells in the signature. To avoid clutter in the list of cells in the signature, instead of inputting higher-level cells indicating invertibility of a cell g, a check box could be ticked and these are implicitly added.

4.2 Algorithms

In this section we discuss the type system used to model the signature and diagram structures, as well as the algorithms that implement the operations on them. The data carried by the datatypes corresponding to signatures, diagrams and embeddings is the same as defined in Chapter 2. The sole difference is that for each embedding we omitted the information about its domain and codomain, instead leaving us with just an array of natural numbers, which in this section we refer to as *coordinates*.

First, we establish the datatypes, let:

- $\mathrm{Sig}(n)$ be the datatype that implements the signature structure as described in Definition 2.2.1.
- $\mathrm{Diag}(n, \sigma)$ be the datatype that implements the diagram structure as described in Definition 2.2.2.

- $\mathrm{Emb}(S, D)$ be the datatype that implements the embedding structure as described in Definition 2.2.7.

Globular encodes all of these as combinatorial data in accordance with the respective definition of each structure. In this presentation, we distinguish between two categories of algorithms that implement the theoretical setup. The first category is algorithms that directly correspond to the operations of composition and rewriting of diagram structures. The other category is auxiliary algorithms that are related to the user interface. In the following exposition we provide a thorough discussion of the former and an intuitive description of the latter.

4.2.1 Core algorithms

Description of each procedure consists of specifying the input and output types, as well as the individual steps transforming the former into the latter. Correctness for all core operations has already been proved in Chapter 2. For the purpose of analysing computational complexity, let us first define the following as the size of the diagram structure:

Definition 4.2.1. Given an n-diagram D, the *size* of the structure, which we denote by $\Sigma(D)$, is:

- If, $n = 0$, then: $\Sigma(D) = 1$
- If, $n > 0$, then: $\Sigma(D) = n * |D| + \Sigma(D.s)$

Intuitively, this corresponds to the number of cells in D and all of its sources multiplied by the maximum length of the component embedding for each cell. For each n-diagram, there are $|D|$ generator cells and $|D|$ embeddings, each embedding consists of a list of $(n-1)$ numbers. The size of the diagram is the sum of products, for D and all of its sources, of the total length of the list of generators multiplied by the number of numerical entries in the lists of numbers modelling the embeddings. In most procedures, we need to process each item in these lists at most once.

Matching

$$\mathrm{Match}\big(D : \mathrm{Diag}(n, \sigma), D' : \mathrm{Diag}(n, \sigma)\big) : \{\mathbf{true}, \mathbf{false}\}$$

This procedure is used to determine whether two combinatorial encodings represent the same diagram, it is the implementation of the notion of diagram equivalence, as described in Definition 2.3.7. For two n-diagrams D, D' we first recursively compare whether their sources $D.s$ and $D'.s$ match. If not, there is no need to compare their lists of n-cells and the procedure returns **false**. Otherwise, we compare the lists of generators and embeddings. If they are of different lengths, *i.e.* $|D| \neq |D'|$, return **false**. If $|D| = |D'|$, then if there is an integer $0 \leq k \leq |D|$ such that either types $D[k].g \neq D'[k].g$ or embeddings $D[k].e \neq D'[k].e$ do not match, then return **false**, otherwise return **true**.

In a single recursive call, we process every entry in the list of generators and embeddings of D at most once. As the entire recursive procedure processes D and all its iterated sources, we obtain running time linear in the size of the diagram.

Identity

$$\mathrm{Identity}\big(D : \mathrm{Diag}(n, \sigma)\big) : \mathrm{Diag}(n+1, \sigma)$$

Given an n-diagram D, this operation transforms it into an identity $(n+1)$-diagram $D'(n+1, \sigma)$. It is the implementation of the notion of the identity of a diagram $\mathrm{Id}(D)$, as described in Definition 2.4.22. The list of generators of $\mathrm{Id}(D)$ is empty, while the field $\mathrm{Id}(D).s$ is set to D. That way, all the diagram data is constructed and the procedure terminates. We perform a fixed number of assignment, so the procedure works in constant time.

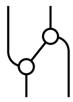
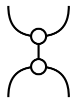
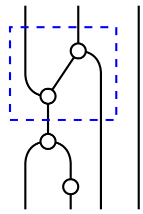
Rewriting

$$\text{Rewrite}\big(D : \text{Diag}(n,\sigma), S : \text{Diag}(n,\sigma), T : \text{Diag}(n,\sigma), e : \text{Emb}(S,D)\big) : \text{Diag}(n,\sigma)$$

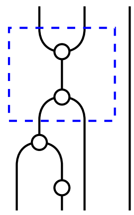
D is the diagram that is being rewritten, S is the source of the rewrite and T its target. The embedding e indicates where in D the rewrite is to be applied. To execute the procedure, $|S|$ consecutive cells in D starting from position $e.h$ are removed, instead cells in T are inserted, with the coordinate in their embeddings $T[i].e$ offset by the numerical values in e.

In the procedure of removing generators of S from D and inserting generators of T instead, every cell is processed at most once. Numerical values in each embedding in T get augmented at most once, hence the procedure is linear in the size of diagrams S and T.

Let us consider a specific example. S is the source of the rewrite and T its target, D is the diagram being rewritten and the location of the rewrite source within D is denoted by the blue dashed rectangle.

S =  T =  D = 

Then the rewritten diagram is as follows, where the rewritten section of the diagram is denoted by the blue dashed rectangle.

$\text{Rewrite}(D,S,T,e)$ = 

Attachment

$$\text{Attach}(D : \text{Diag}(n,\sigma), S : \text{Diag}(k,\sigma), b : \{s,t\}, e : \text{Emb}(s(S), t^{n-k+1}(D)) : \text{Diag}(n,\sigma)$$

This procedure is the implementation of the operation of diagram composition, as described in Definition 2.4.6. The term 'attachment' is used to indicate the effect the procedure has on the diagram in the workspace, where a visual effect of attaching a diagram is created. We attach the diagram S to the diagram D, b is the boolean value indicating whether we are attaching to the source or the target boundary. e is the embedding of the source or target of S in the appropriate source or target of D, with the combination being determined by the value of the boolean b.

The procedure is executed as follows:

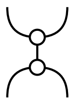
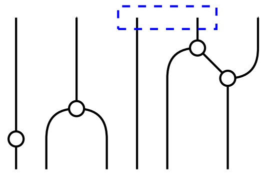
- If $n-k=0$, depending on the value of b, we either append the elements in the lists of generators and embeddings of S at the end ($b=t$) or the beginning ($b=s$) of D's corresponding lists. We use the numerical data in e to offset the coordinates in each $S[i].e$. Additionally, if $b=s$, the source boundary needs to be modified, so it is rewritten using elements $S[i].g$ as rewriting cells.

- If $n-k>0$, the procedure is called recursively for $D.s$, with S, b and e as parameters. After the recursive call concludes, for $0 \leq i \leq |D|$ we augment $D[i].e$ by the offset created by adding new $(n-1)$-cells to $D.s$.

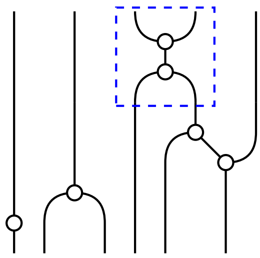
Note that, this procedure corresponds to first implicitly whiskering S, so that its appropriate source or target matches that of D, and then composing S with D in the usual way.

In the procedure, we need to process every element in S at most once, this happens at the moment when the element gets added to the appropriate part of D. In the scenario where S is attached to the source boundary of D, additionally the rewriting procedure needs to be performed $|S|$ times on the source of D. It is this second step which is more costly, hence overall, the number of operations is bounded by the time complexity of performing the additional rewrites, *i.e.* $|S|$ times the size of the diagram $D.s$.

In the example below, S is the diagram being attached, D is the diagram we are attaching to, the boundary and the specific coordinates of the attachment point are illustrated by the blue dashed rectangle.

$S \quad =$  $\qquad D \quad =$ 

The resulting diagram is as follows, where S is denoted by the blue dashed rectangle.

$\text{Attach}(D, S, t, e) \quad =$ 

Slicing

$$\text{Slice}(D : \text{Diag}(n, \sigma), k : \mathbb{N}) : \text{Diag}(n-1, \sigma.\sigma)$$

Given an n-diagram D, we can rewrite the source boundary $D.s$ using the initial k entries in its list of generators in accordance with Definition 2.2.3. This gives us the k-th slice of D. To execute the procedure we rewrite $D.s$, using elements in D's lists of generators and embeddings, k times.

Correctness of this procedure is the direct consequence of correctness of the rewriting procedure. As we perform the procedure of rewriting on an $(n-1)$-diagrams, the procedure takes at most k times the size of the diagram $D.s$. An important note is that the resulting $(n-1)$-diagram may be given as input to another instance of the procedure. This way, we may obtain a slice of D of an arbitrary dimension and location.

Verification

$$\text{Verify}\big(D : \text{Diag}(n, \sigma)\big) : \{\textbf{true}, \textbf{false}\}$$

Given a piece of data D of the type $\text{Diag}(n, \sigma)$, this procedure allows us to verify whether D is in fact a valid diagram, *i.e.* is well-defined in the sense of Definition 2.2.6.

First, the procedure is called recursively for $D.s$, if the call returns **false**, the entire procedure returns **false**. Otherwise, we attempt to rewrite $D.s$ using the generators $D[i].g$ in the list. For each i, first we need to perform a globularity check on diagrams $s(D[i].g)$ and $t(D[i].g)$, which we do using the procedure **Match** on their sources and targets. Then we compare the top coordinate $D[i].e.h$ and check whether it is in range for the size of the slice $D[i].d$. If any of these checks fails, return **false**. Otherwise, rewrite $D[i].d$ into $D[i+1].d$ and repeat the step $|D|$ times.

If all these checks succeeded, D is a well-defined diagram and the procedure returns **true**. The procedure rewrites and matches $(n-1)$-diagrams at most $|D|$ times starting with the diagram $D.s$, hence overall it takes $|D|$ times the size of diagram $D.s$.

Construct Interchanger

$$\text{Interchanger}\big(D : \text{Diag}(n,\sigma), \Psi_n, h\big) : (\text{Diag}(n,\sigma), \text{Diag}(n,\sigma))$$

The procedure takes a diagram D, an interchanger type Ψ and coordinates e as input and returns the source diagram and the target diagram of the instance of Ψ in D at location h, which allows to use interchangers in rewriting in a systematic way.

The source and the target for each variant of every interchanger type is constructed manually in accordance with definitions listed in Chapter 3. S and T constructed in such a way describe how the combinatorial description of the diagram D changes when an interchanger of type Ψ gets applied at location h.

An important point is that for an application of a non composite interchanger of types I-VI, it is sufficient to provide a single integer h within a diagram. This is interpreted as the height which the highest-level cell involved in the move is located at. Given an n-diagram, an interchanger of type Ψ_n acts on n-cells within D. Since these n-cells are organised in the list $D.g$ we have the following:

- For interchangers of type I, we adapt a convention and specify the location of the cell that appears earlier in the list $D.g$, this is sufficient as the cells being interchanged have to be adjacent.
- For interchangers of types $\text{II}, \text{III}, \text{IV}$ that capture naturality in one variable for some lower-level interchanger cell, it is sufficient to specify the location of the individual cell that is subject to naturality.
- For interchangers of type V that capture naturality of I_2 in two of its variables, we adapt a convention and specify the location of the cell that appears earlier in the list $D.g$, this is sufficient as both cells have to be adjacent.

4.2.2 User interface algorithms

The second category of algorithms are those that are not directly related to the properties of diagram structures proved in Chapter 2. They are non-essential with regards to the correctness of the approach, however they do make the tool significantly easier to use.

Enumeration

$$\text{Enumerate}\big(S : \text{Diag}(n,\sigma), D : \text{Diag}(n,\sigma)\big) : \text{List}(\text{Emb}(S,D))$$

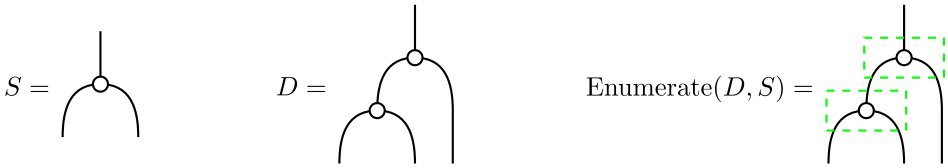
Given two n-diagrams S, D, this procedure lists all the individual instances of S being a subdiagram of D, or in other words, lists all the embeddings of S in D.

First we want to find a platform for the match, *i.e.* an index h such that $S[0].d$ is a subdiagram of $D[h].d$. For this we call the procedure recursively for $S[0].d$ and $D[h].d$. Given a list of embeddings of $S[0].d$ in $D[h].d$ there are two possibilities.

- If the list $S.g$ is non-empty, we select the unique embedding consistent with the source of the generator $D[h].g$, let us refer to it as e'. We then proceed to comparing elements $S[j].g$ and $S[j].e$ with $D[h+j].g$ and $D[h+j].e$. If any of these checks return a mismatch, the embedding is discarded. Otherwise, an embedding of S in D has been found and we append h to the list of numerical values of e' to obtain the embedding e. Since, we are interested in finding all embeddings of S in D, the procedure is repeated for all $0 \leq h \leq |D| - |S|$.
- If the list $S.g$ is empty, then we promote all the embeddings of $S[0].d$ in $D[h].d$ to embeddings of S in D by appending h to the list of numerical values for each embedding.

For every recursive call, for n-diagrams S, D the procedure conducts at most $|D| - |S|$ matching operations on diagrams and calls itself recursively $|D| - |S|$ times on an $(n-1)$-diagram. In the worst case scenario, when an n-diagram S consists of a single 0-cell, that results in the running time exponential in the size of D and S. However, for an n-diagram S whose list of generators is non-empty, after each recursive call, we only select one match consistent with the structure of D. This ensures that the running time is polynomial in the size of D and S.

We illustrate enumeration with the following example. Let us consider two diagrams: D and S:



If the returned list is non-empty, we can infer that S is a subdiagram of D. The procedure of enumeration is used as pre-processing step for rewriting and attachment to obtain the embedding that needs to be supplied as the input for each of these procedures. If more than one option is available for the given pair of selected diagrams, enumeration enables the user to select the attachment point or the section of the diagram to be rewritten,

As discussed above for rewriting, for a diagram D and a rewrite defined by S and T, enumeration looks for embeddings of S in D. For attachment, for a diagram S being attached to D, enumeration looks for embeddings of $s(S)$ in the appropriate target of D *and* embeddings of $t(S)$ in the appropriate source of D. Selection of an embedding of one of these types additionally supplies the boolean indicating whether S is being attached to the source or to the target of D, which is a required input for attachment.

Layout

This is the procedure that given an n-diagram D, transforms it into the graphical representation G_D in accordance with Definition 3.1.1. For $n = 1$ this representation is a string of dots on a line and for $n = 2$, it is a string diagram. As at the moment, Globular does not support graphics for $n = 3$ *i.e.* surface diagrams embedded in $\mathbb{R}^3$, instead projections of $\mathbb{R}^n$ to $\mathbb{R}^2$ are used for higher dimensions. These projections can then by manipulated from the user's interface by a set of toggles, as discussed in Section 4.1.

Before the graphical representation is rendered on the screen there is an intermediate step of pre-processing the combinatorial representation of the n-diagram D to obtain:

- Planar x, y coordinates of each vertex representing an n-cell
- Planar x, y coordinates of both endpoints for each straight line representing an $(n-1)$-cell
- Planar coordinates of the bottom left and top right corner for each rectangle representing an $(n-2)$-cell

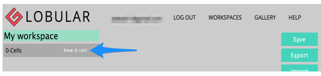
Coordinates for graphical representations of all $(n-1)$-slices of D are produced recursively, which is possible given that the bottom $(n-2)$ dimensions are projected out. The coordinates for the combined 2D graphical representation of D are then obtained as described in the example presented in Section 3.1.

4.3 Using Globular

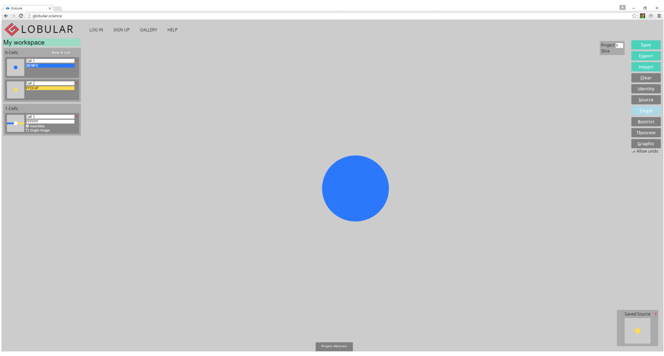
Constructing a theory and proving theorems in Globular is an inductive process, whereby lower-dimensional objects are used to construct higher-dimensional objects. Since the diagram and signature structures are mutually dependent, this is achieved by building up the signature in parallel with increasingly higher-dimensional diagrams. When the webpage hosting the tool is loaded the starting screen is as follows:

The signature menu is on the left hand side of the screen, the action menu on the right. The central area is the main workspace. Note that the pre-loaded signature consists of a single 0-cell. The user has a choice of either loading one of their pre-saved projects, opening one of the publicly available projects from the gallery, or building a signature from scratch.

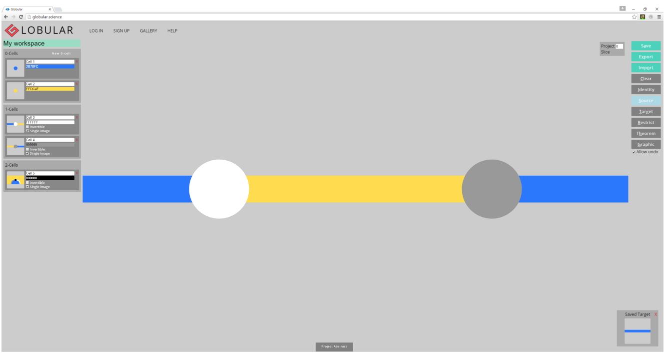
Here, we choose the final option and we recreate the signature used to prove Theorem 1.3.4 on promoting an equivalence to an adjoint equivalence in a 2-category. To build a new project from an empty 0-signature, the first step is to add new 0-cells:



The newly added 0-cells can now become the sources and targets of a new 1-cell. This is achieved by first selecting them from the signature menu to appear as diagrams in the main workspace and then designating them as the source or target by clicking the corresponding button in the action menu.

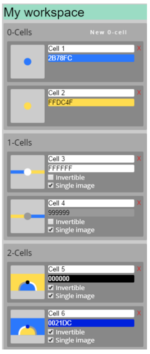


Here, we create 1-cells $A \xrightarrow{F} B$ and $B \xrightarrow{G} A$ that are going to witness the equivalence that is being constructed. As per Definition 2.4.6, diagrams corresponding to 1-cells can be *composed* with other diagrams. This allows us to form composite sources and targets for 2-cells.



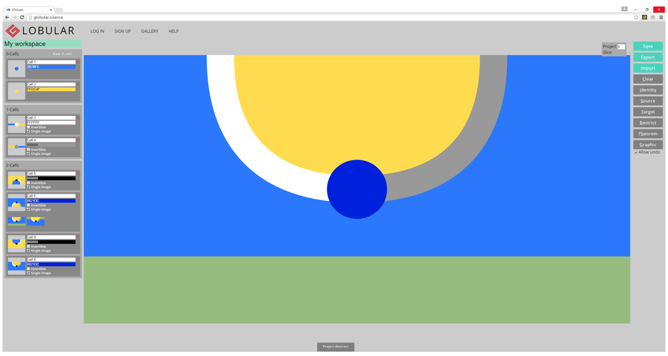
In this step, 2-cells $F \circ G \xrightarrow{\alpha} \mathrm{id}_A$ and $\mathrm{id}_A \xrightarrow{\beta} G \circ F$ are created. For k-cells with $k \geq 2$, when both the potential source and target are selected, Globular conducts a type check whether the cells satisfy the globularity conditions and can indeed be made into a $(k+1)$-cell.

Finally, to complete the description of the equivalence, we need to add equations that capture that α and β are invertible. There are two methods in which this could be obtained. The first is to check the invertibility boxes for both cells:

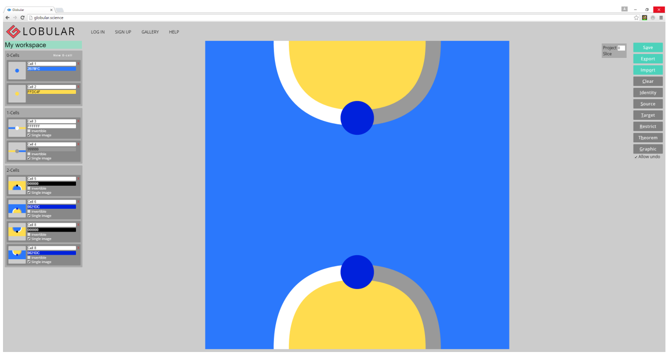


This automatically adds all higher level cells that are needed to make the given cell invertible in the sense of Definition 3.2.7. This is done implicitly, without listing these cells in the signature menu, as to avoid clutter. An alternative method is to input them explicitly, which is the path we follow here. When the workspace is empty and an element of the signature is selected, its identity diagram is created in the workspace. This makes use of the inclusion function $G_k \to {\Delta^*}_k$ as described in Definition 2.2.4.

If the workspace is non-empty, selecting an element of the signature triggers a launch of the enumeration procedure, that looks for all the positions where the new element can be attached. All matches are then listed in the signature menu and highlighted when hovered over.



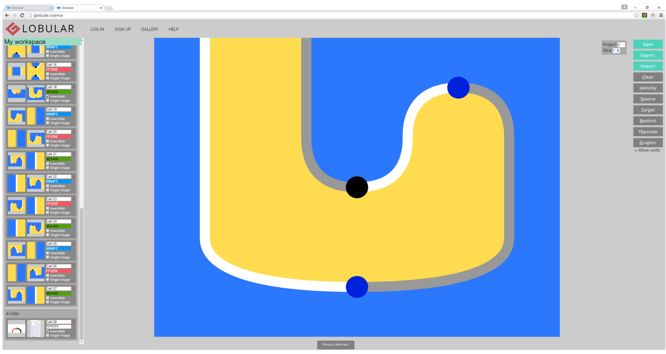
Once one of the matches is selected, the procedure of composition attaches the new element to the diagram in the workspace.



By this process, a signature capturing the concept of equivalence in a 2-category is built:

If one wants to construct a signature to work in a quasistrict 3- or 4-category, the procedure of building elements of the signature could continue by adding higher-level cells up to $n = 5$ (equations for a quasistrict 4-category).

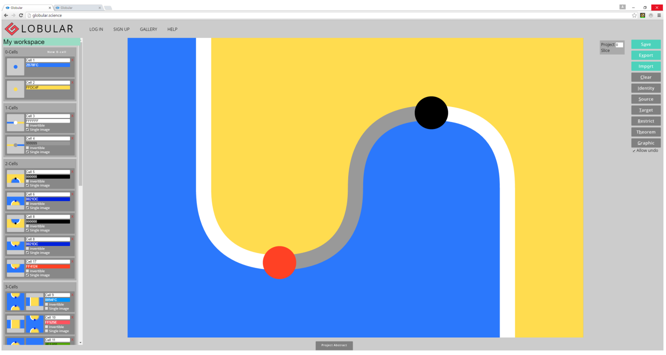
We now define a new 2-cell α', that together with β will witness an adjoint equivalence. This is achieved using the 'Theorem' functionality. After a desired new 2-cell has been created in the workspace from the existing generators, we click the 'Theorem' button in the action menu. This results in first, a fresh 2-cell being added to the signature and second, adding a 3-cell that has the new generator as its source and our desired new composite cell as its target. That way, this new cell serves as the *definition* of the new composite cell being added to the signature.



We can now prove that this new 2-cell satisfies the snake equations. As discussed in Chapter 1, proofs in Globular are viewed as sequences of rewrites, hence to prove that an equation is satisfied means to show that the left hand side of an equation could be rewritten into the right hand side by a series of (3-cell) rewrites.

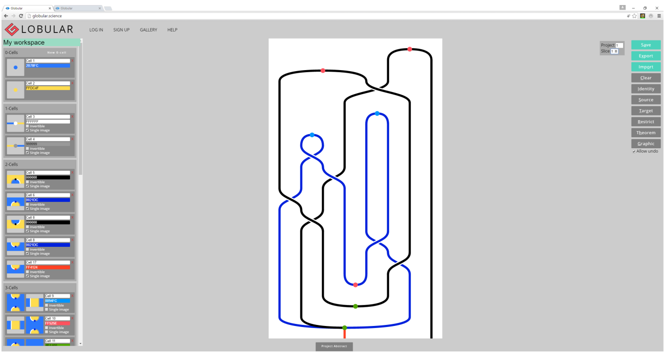
Note that rewriting a 2-diagram D by a 3-cell α could be viewed as the same as first creating the *identity* diagram on D and then composing it with α. That way, the source of $\mathrm{Id}(D) \circ \alpha$ is D and its target is the rewrite of D. In this perspective a 3-cell proof is a history of rewriting one 2-cell into another, while recording all intermediate stages.

To show that the first snake equation is satisfied, we build the 2-diagram that corresponds to its left hand side and create an identity 3-diagram, by selecting the appropriate button in the action menu.



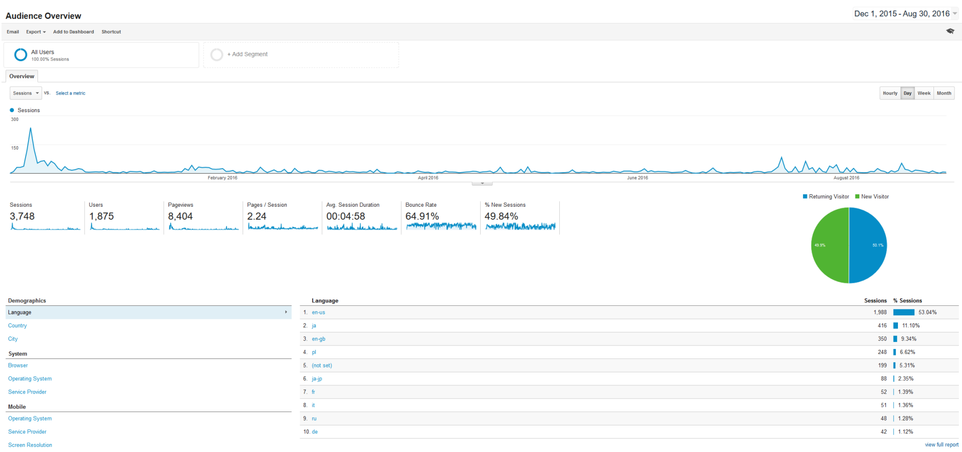
The *project* and *slice* controls can be used to navigate the proof, they are both placed at the top-right of the workspace. Since currently Globular only supports 2D graphics we need to make use of the projection property of graphical representations, as described in Definition 3.1.1. Now we could create the same derivation as used in the proof of Theorem 1.3.4

For an identity 3-cell, the view is automatically set to its final slice, so that a sequence of rewrites can be easily started. The proof can be then constructed by either attaching single step rewrite 3-cells from the signature, or attaching one of the special *interchanger* cells, as discussed in Section 4.1. Finally, the entire proof can be viewed as a single 4-cell in the projected view:



4.4 Other examples

Since its launch in December 2015 Globular has attracted a community of scholars to regularly use it as a proof assistant. Overall, in 9 months, the webpage has been visited over 8000 times by 1800 unique users, with 800 visits by 200 unique users coming in the month of August 2016.



These users created 36 publicly available proofs and worked on many more in their private accounts. To conclude this chapter, we give several examples of proofs from algebra and topology formalised by the community. For each of them, we briefly describe the mathematical context of the proof, and give some details of its formalisation. We use the hyperlinking feature provided by Globular and link directly to the formalised proofs on the website. To the best of our knowledge, none of these results have previously been formalised by any existing tool.

Example 4.4.1 (Frobenius implies associative, globular.science/1512.004, length 12 steps)**.** *In a monoidal category, if multiplication and comultiplication morphisms are unital, counital and Frobenius, then they are associative and coassociative. We formalise this in Globular using a 2-category with a single 0-cell, since, as discussed in Section 1.2.3, this is algebraically equivalent to a monoidal category. Such a proof would be traditionally written out as a series of pictures; for example, see the textbook [34]. Globular produces these pictures automatically.*

Example 4.4.2 (Strengthening an equivalence, globular.science/1512.007, length 14 steps)**.** *In a 2-category, an equivalence gives rise to an adjoint equivalence. This is a classic result in category theory [6, 48]. It can be considered as one of the first non-trivial theorems of 2-category theory. We presented a graphical proof based on the formalisation in Globular in Section 1.3.*

Example 4.4.3 (Swallowtail comes for free, globular.science/1512.006, length 12 steps)**.** *In a monoidal 2-category, a weakly-dual pair of objects gives rise to a strongly-dual pair, satisfying the swallowtail equations. This theorem plays an important role in the singularity theory of 3-manifolds [45]. In the formalisation, we again make use of the periodic table of higher categories and model a monoidal 2-category as a 3-category with one 0-cell.*

Example 4.4.4 (Pentagon and triangle implies $\rho_I = \lambda_I$, globular.science/1512.002, length 62 steps)**.** *In a monoidal 2-category, a pseudomonoid object satisfies $\rho_I = \lambda_I$. A pseudomonoid is a higher algebraic structure categorifying the concept of monoid; it has the property that a pseudomonoid in* **Cat** *is the same as a monoidal category. Such a structure is known to be coherent [35], in the sense that all equations commute, and here we give an explicit proof of the equation $\rho_I = \lambda_I$, which played an important role in the early study of coherence for monoidal categories.*

Example 4.4.5 (The antipode is an algebra homomorphism, globular.science/1512.011, length 68 steps)**.** *For a Hopf algebra structure in a braided monoidal category, the antipode is an algebra homomorphism. Hopf algebras are algebraic structures which play an important role in representation theory and physics [39, 56]. Proofs involving these structures are usually presented in Sweedler notation, a linear syntax which represents coalgebraic structures using strings of formal variables with subscripts; we do not know of any existing approaches to formal verification for Sweedler proofs. This formalisation in Globular is translated from a Sweedler proof given in [43]. For the formalisation, we model a braided monoidal category as a 3-category with one 0-cell and one 1-cell.*

Example 4.4.6 (The Perko knots are isotopic, globular.science/1512.012, length 251 steps)**.** *The Perko knots are isotopic. The Perko knots are a pair of 10-crossing knots stated by Little in 1899 to be distinct, but proven by Perko in 1974 to be isotopic [44]. Here we give the isotopy proof, adapted from [40]. A feature worth noting is that the second and third Reidemeister moves do not have to be entered explicitly, since they are already implied by the 3-category axioms (as instances of interchangers of types I_2') and II_3) respectively. The proof consists of a series of 251 atomic deformations, which rewrite the first Perko knot into the second. By stepping through the proof one rewrite at a time, the isotopy itself can be visualised as a movie, in accordance with visualisation rules described in Chapter 3.*

Chapter 5

Adjunctions in higher categories

In a weak n-category, two objects are considered 'the same' or equivalent if there exists a morphism between them which is invertible in a maximally weak sense, *i.e.* up to all higher-level equivalences, as described in Definiion 3.2.7. If equivalence between objects does not arise, the weaker notion of adjunction could be used to reason about related cells. Of particular interest are adjunctions which in some sense carry more structure, *i.e.* are more coherent. It is generally expected that an adjunction of 1-morphisms in a weak higher category gives rise to a more coherent adjunction. This was shown by Verity and Riehl [47] for an adjunction in an arbitrary weak n-category, however assuming the homotopy hypothesis [5, 46] and without providing an explicit construction.

In this chapter, we prove two results on promoting adjunctions in a 3-category and in a quasistrict 4-category. The first is Theorem 5.0.2 on an adjunction in a 3-category giving rise to a coherent adjunction satisfying the swallowtail equations, which was first proved by Verity [59] and later discussed in depth by Gurski [26, 27]. The second is Theorem 5.0.3 on an adjunction in a quasistrict 4-category giving rise to a coherent adjunction satisfying the butterfly equations. This is a result that, to the best of our knowledge, is the first substantial proof of a non-trivial property conducted explicitly in the setting of a 4-category. Both proofs have been formalised with the aid of Globular. This has been of particular assistance for the proof of Theorem 5.0.3, as the entire derivation for just one of the two butterfly equations consists of 140 5-cell rewrites. Recall the intuition for visualising higher-level cells described in Section 3.1, according to the observations made there, an individual 5-cell is a method of rewriting a 'movie' of 3D geometrical objects into another movie. Hence, the entire derivation, as a sequence of 5-cells, is a movie of movies of 3D objects. If one was to carry out a more traditional derivation and only use 2D structures, then we have one more level of complexity and obtain a movie of movies of movies of 2D geometrical objects. An entire derivation would then consist of several thousand 2D geometrical objects appropriately organised to form higher-level cells. We believe that developing such a large structure would be challenging without software assistance.

While a more general result on promoting adjunctions in weak n-categories has been proved by Verity and Riehl [47], there are several reasons why our result is still significant. Firstly, the approach by Verity and Rieh uses $(\infty, 1)$-categories built on simplices. Due to this, the result is dependent on the homotopy hypothesis and the association between ∞-groupoids and topological spaces. This is a strong assumption which is expected to be difficult to prove. Secondly, it may be argued that for the purposes of giving further evidence for correctness of the definition of a quasistrict 4-category, it is actually more suitable to prove a result that is expected to be true. If unsuccessful, we perhaps would have been able to identify weaknesses in the chosen approach. Finally, there is still a substantial value in providing explicit proofs of categorical facts, especially if the proof is carried out in a 4-categorical setting in which no other comparable proof has been given in the literature.

Recall Definition 1.3.2 of an adjunction in a 2-category. It generalises to weak n-categories for $n \geq 3$. The main difference is that for $n \geq 3$, equality between 3-cells becomes equivalence in the maximally weak sense, so we get more higher-level structure. Also, recall Definition 3.2.7 of what it means for a

k-cell in an n-signature to be invertible.

Definition 5.0.1. In a weak n-category, an *adjunction* is a pair of objects A and B, a pair of 1-cells $A \xrightarrow{F} B$ and $B \xrightarrow{G} A$, a pair of 2-cells $F \circ G \xrightarrow{\alpha} \mathrm{id}_A$ and $\mathrm{id}_A \xrightarrow{\beta} G \circ F$, and a pair of invertible 3-morphisms Φ, Ψ such that Φ, Ψ are defined as follows:

The colourfully-named swallowtail and butterfly equations derive their names from the geometrical shapes that their graphical representations resemble. Additionally, they are connected to the swallowtail and butterfly singularities in the classification of catastrophes in the theory due to Thom [58]. These are part of a deeper and not, as of yet, well-understood connection between n-categories admitting adjunctions and singularity theory. The theory, which is concerned with non-linear systems in which small variations in input variables cause equilibria to emerge or disappear, is itself deeply fascinating, to the extent that it attracted admirers from outside of the community of mathematicians and inspired Salvador Dali's final painting.

In this investigation, we concentrate on adjunctions in a weak 3-category and in a quasistrict 4-category. An important note is that if Conjecture 3.1.2 holds, then Theorem 5.0.3 immediately holds in any weak 4-category. Recall that in Chapter 4, we illustrated what higher-level coherences of types III_4, IV_4 and V_4 look like graphically in a projected 2D notation. In the derivations below, we make extensive use of these interchangers, including slices of lower level projections where appropriate.

Before we proceed to proving the main result of this chapter, we first show an auxiliary result which, as mentioned above, was first proved by Verity [59]. Recall from Chapter 1 that equations are given here by invertible higher-level cells, hence a proof of equality is given by a sequence of rewrites relating one cell to the other.

Theorem 5.0.2. *An adjunction of 1-morphisms in a 3-category gives rise to a coherent adjunction satisfying the swallowtail equations.*

Proof. By Theorem 3.6.2, it is sufficient to show this in the setting of a quasistrict 3-category as described in Definition 3.2.2. Let objects A, B, 1-morphisms $A \xrightarrow{f} B$, $B \xrightarrow{G} A$, 2-morphisms $f \circ g \xrightarrow{\alpha} \mathrm{id}_A$, $\mathrm{id}_A \xrightarrow{\beta} f \circ g$ and invertible 3-morphisms Φ, Ψ witness an adjunction in a quasistrict 3-category. Let α, β be as follows:

$\alpha \quad =$  $\qquad \beta \quad =$ 

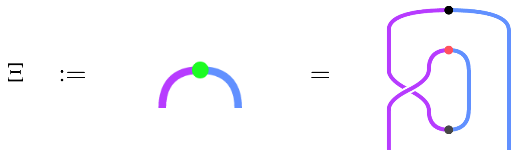
Let Φ, Ψ be defined as follows:

Equivalently, they could be expressed as a single image in the projected view:

$\Phi \quad =$  $\qquad \Psi \quad =$ 

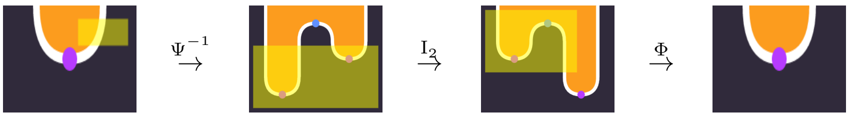
Their inverses are obtained by flipping both diagrams about the x-axis. We denote inverses by colouring the vertices in a slightly darker shade of the colour of the original cell. First let us define an alternative formulation for the first snake equation. This is as follows:

In a projected graphical representation this is expressed as:

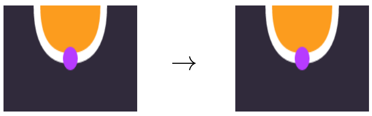


The new 3-cell is clearly invertible, as it is built from invertible morphisms Φ and Ψ. Then the swallowtail equations for Ξ and Ψ take the following form:

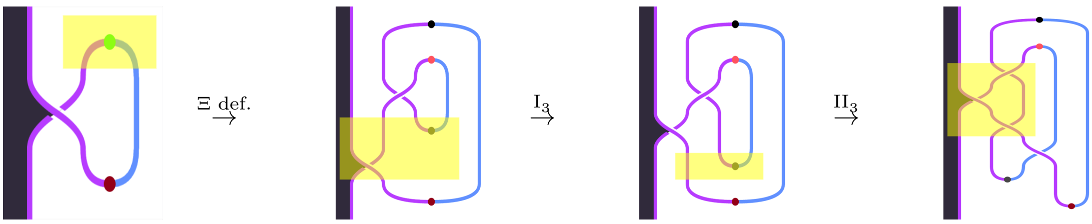
The first equation looks as follows in the unprojected view, its source is:



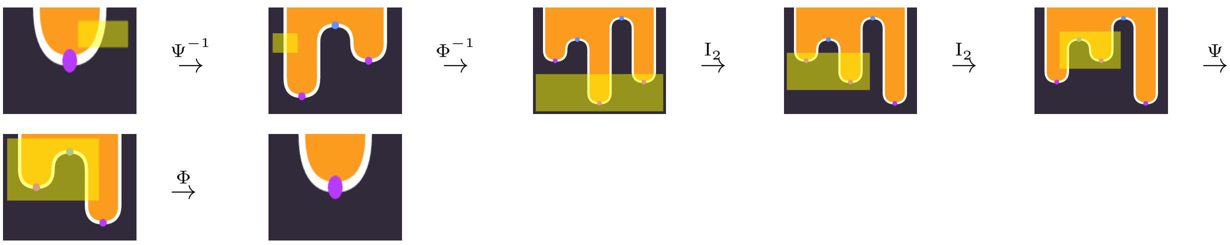
Its target is:



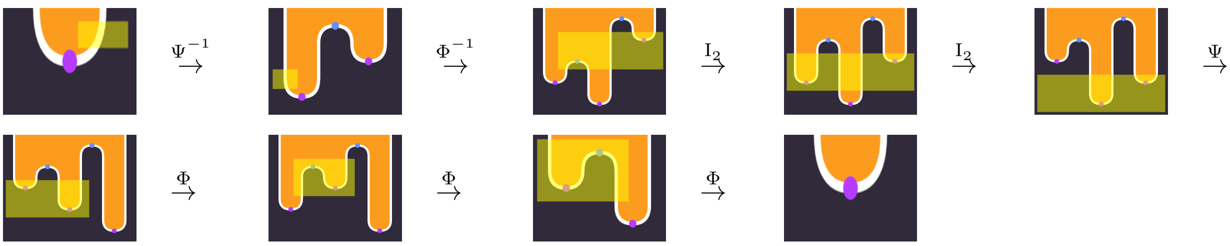
Below, we present a series of rewrites which proves that the first swallowtail equation is satisfied, we begin by showing the first three moves:



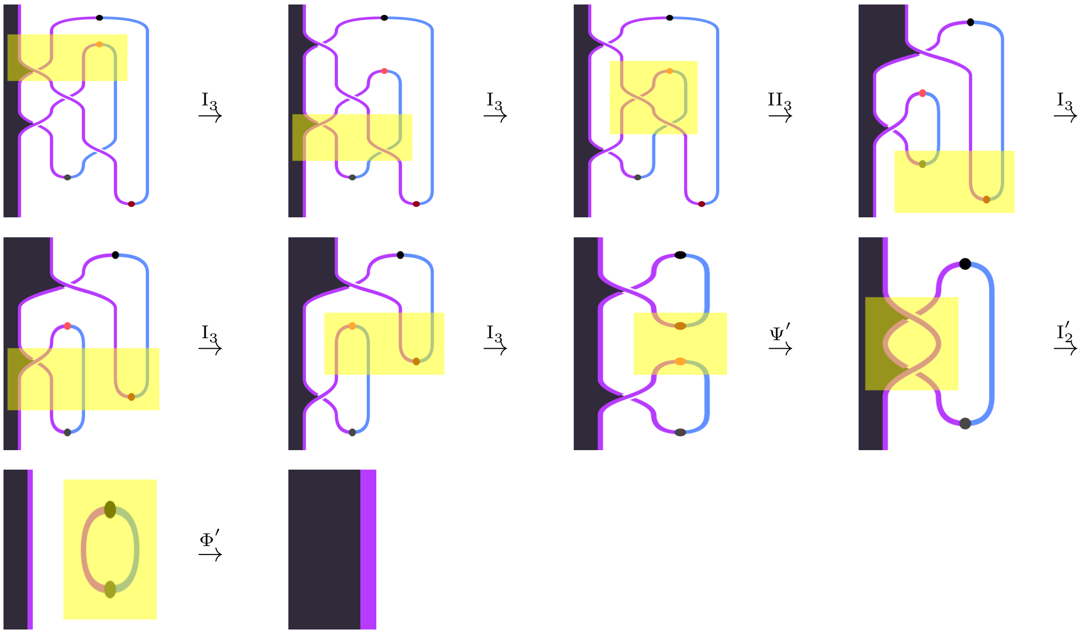
Let us express the last 4-cell in this sequence in the unprojected view. The source is as follows:



The target is:



Note that after the interchanger of type II is performed, the cusp gets introduced below the middle purple cell β. Due to this fact more interchangers of type I_2 have to be applied to swap the heights of vertices. We pick up the main derivation again:



In the final three steps invertibility of both cusps and of the interchanger of type I_2 is used. By this derivation, the first swallowtail equation is satisfied. The other equation follows in a similar manner, hence the adjunction witnessed by objects A, B, 1-morphisms $A \xrightarrow{f} B$, $B \xrightarrow{G} A$, 2-morphisms $f \circ g \xrightarrow{\alpha} \mathrm{id}_A$, $\mathrm{id}_A \xrightarrow{\alpha} f \circ g$ and invertible 3-morphisms Φ', Ψ satisfied the swallowtail equations, as required. $\square$

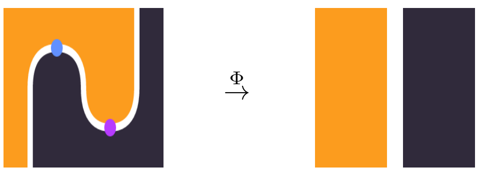
This can now be used to show the main result in this chapter, the theorem on promotion of adjunctions in a quasistrict 4-category. The entire derivation consists of 140 steps, which we break up into manageable parts. In the instances where a composite interchanger is applied, a single cell is used to illustrate multiple steps.

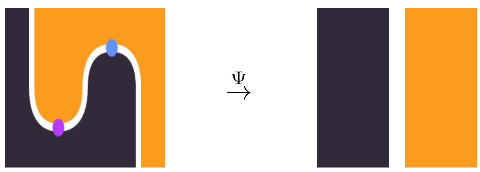
Theorem 5.0.3. *An adjunction of 1-morphisms in a quasistrict 4-category gives rise to a coherent adjunction satisfying the butterfly equations.*

Proof. We are given an adjunction of 1-morphisms in a quasistrict 4-category witnessed by objects A, B, 1-morphisms $A \xrightarrow{f} B$, $B \xrightarrow{G} A$, 2-morphisms $f \circ g \xrightarrow{\alpha} \mathrm{id}_A$, $\mathrm{id}_A \xrightarrow{\alpha} f \circ g$ and invertible 3-morphisms Φ, Ψ witness an adjunction in a quasistrict 3-category. Let α, β be as follows:

$\alpha \quad =$  $\qquad \beta \quad =$ 

Let Φ, Ψ be defined as follows:



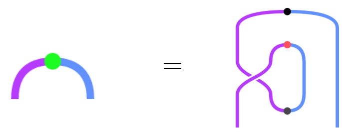


Equivalently, they could be expressed as a single image in the projected view:

$\Phi \quad =$  $\qquad \Psi \quad =$ 

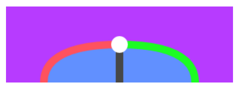
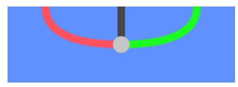
Their inverses are obtained by flipping both diagrams about the x-axis. We denote inverses by colouring the vertices in a slightly darker shade of the colour of the original cell.

By Theorem 5.0.2 we obtain a coherent adjunction of 1-morphisms satisfying the swallowtail equations witnessed by the same morphisms with Φ substituted by Ξ, defined as follows:

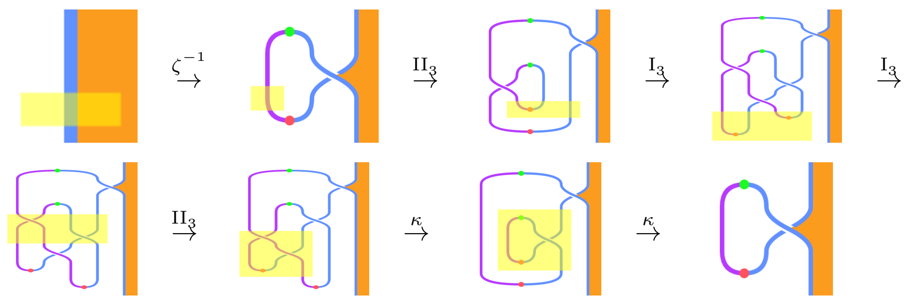
Ξ := 

The new 3-cell is clearly invertible, as it is built from invertible morphisms Φ and Ψ. Then the swallowtail equations for Ξ and Ψ take the following form:

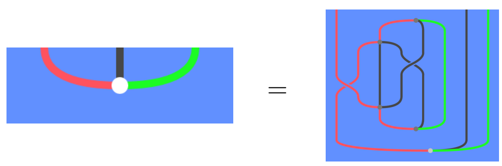
Note that, since we now work in a 4-category, these 4-cells no longer play the role of equations, instead we refer to them as swallowtailator 4-cells. Alternatively, they could be presented as single pictures in the projected graphical representation:

ζ =  κ^{-1} = 

First let us define an alternative formulation for κ^{-1}, which is the inverse of second 4-cell swallowtailator. This is as follows:

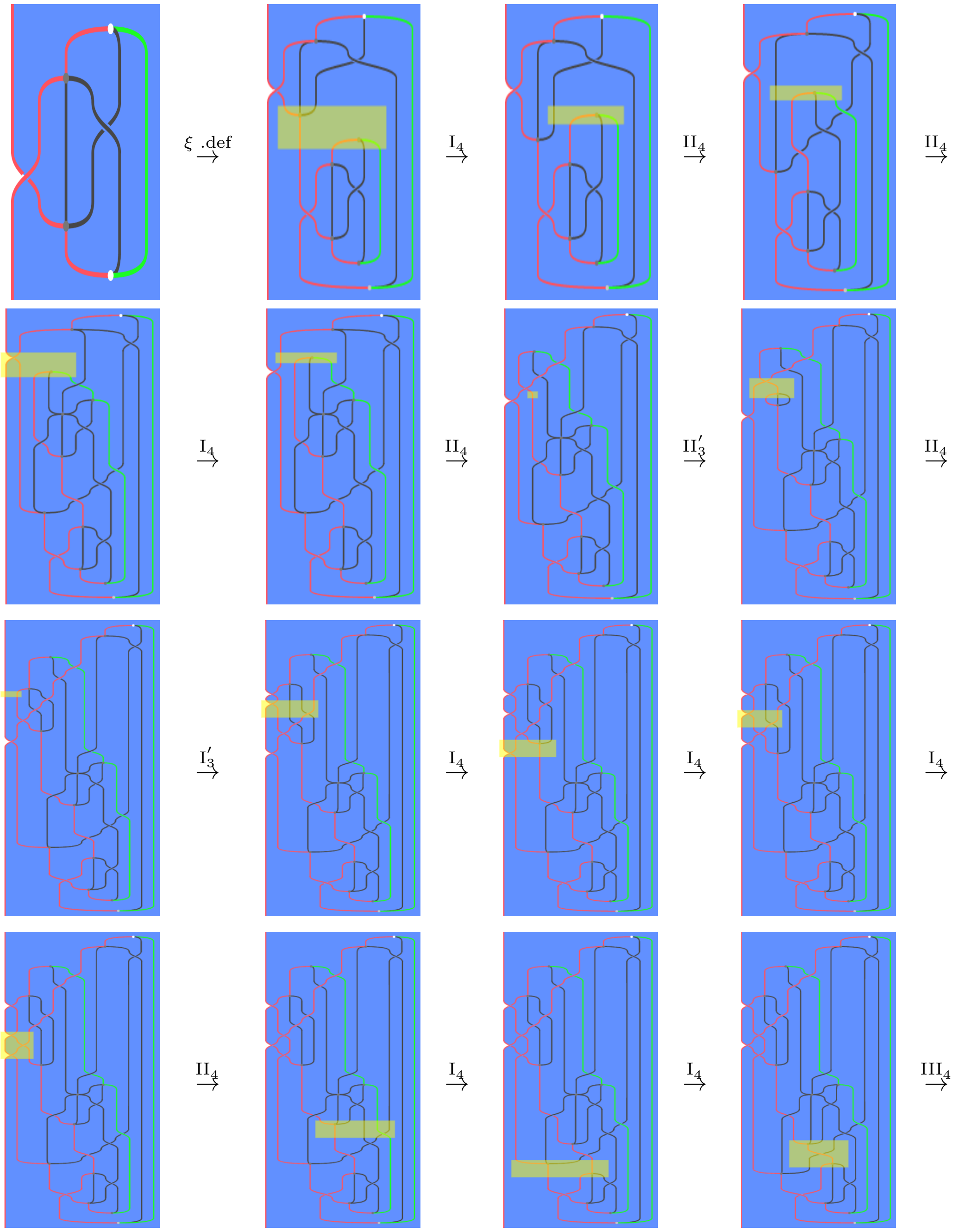


In a projected graphical representation this is expressed as:

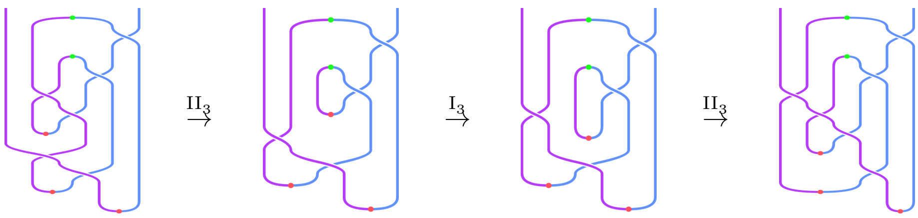
ξ := 

As all morphisms used to build this 4-cell are invertible, so is this cell. The butterfly equations for the first swallowtailator cell and the newly defined cell are expressed by the following 5-cells:

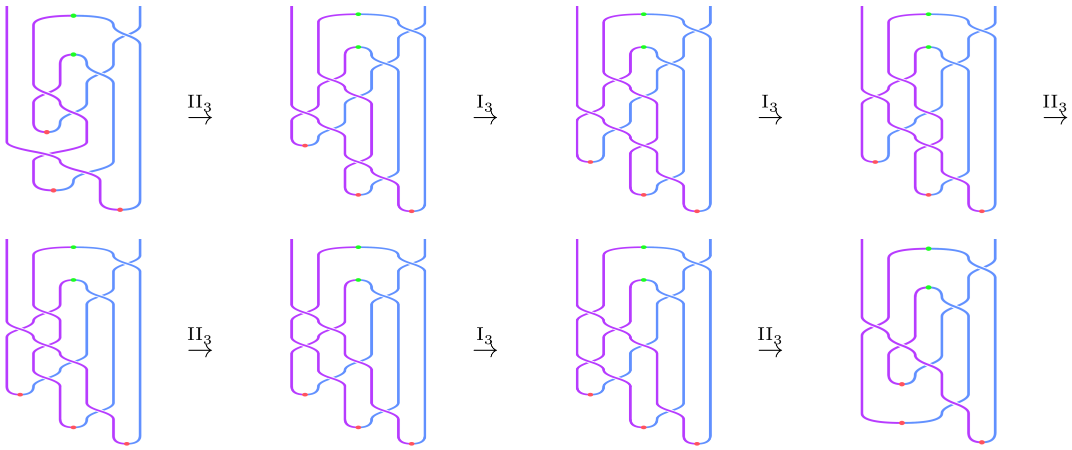
Below, we present a series of rewrites which proves that the first butterfly equation is satisfied:



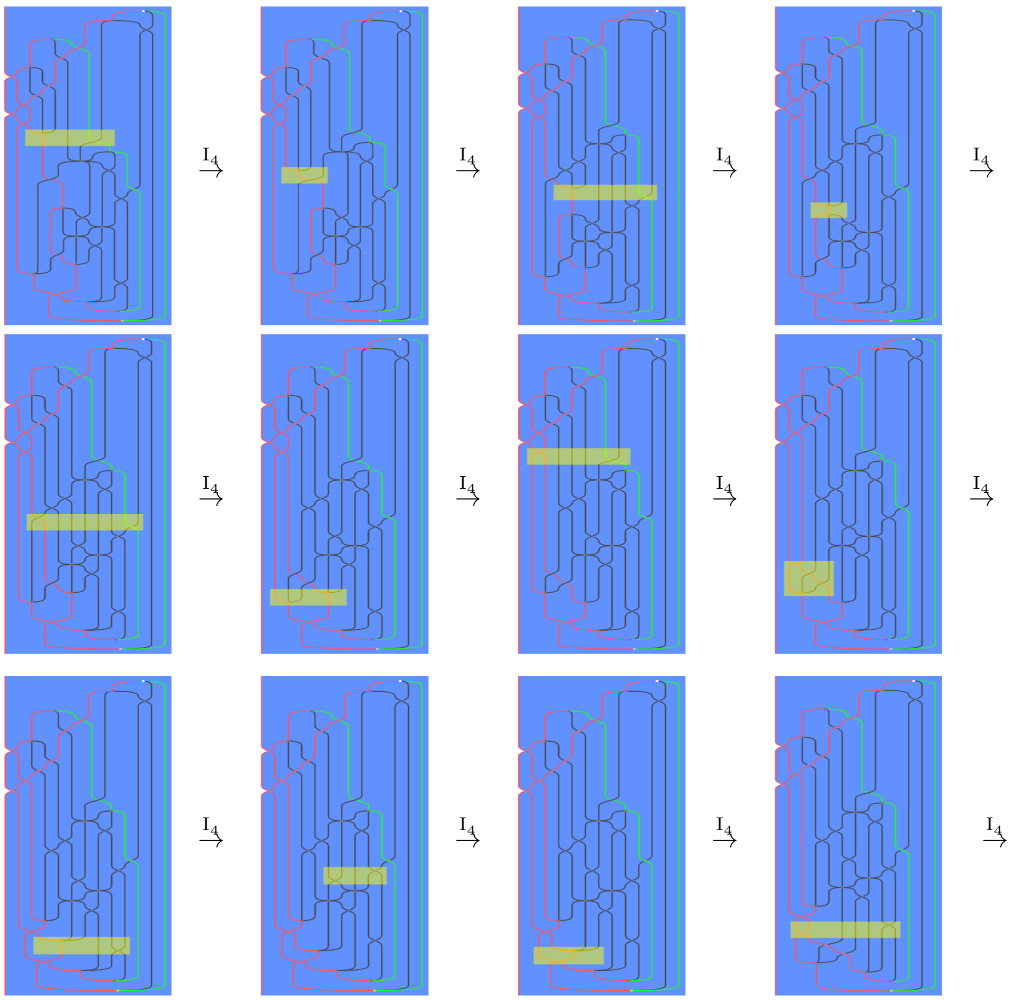
To visualise the next step, we illustrate the last slice rendered above in unprojected form, in the neighbourhood of the highlight box indicating the source of the next rewrite::

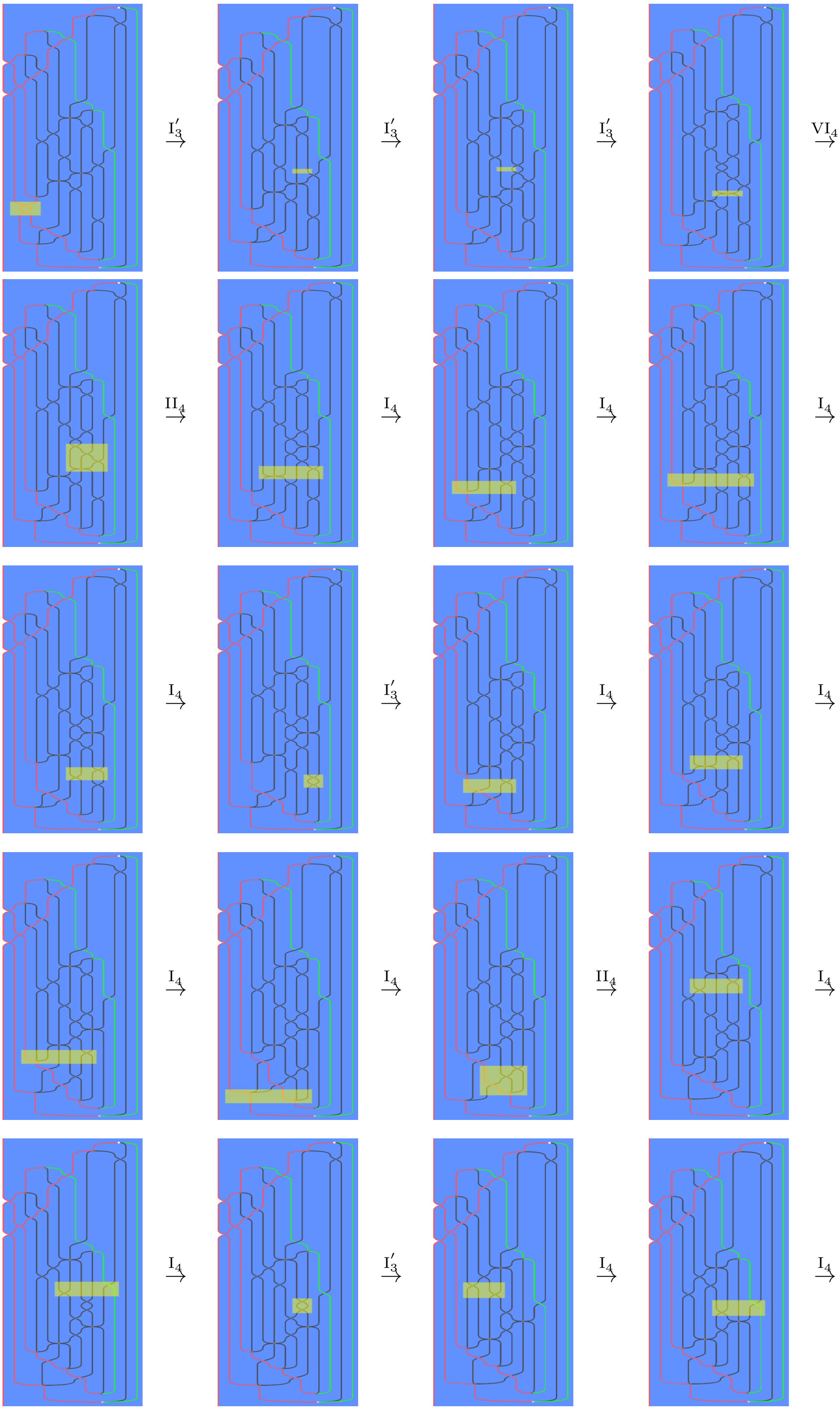


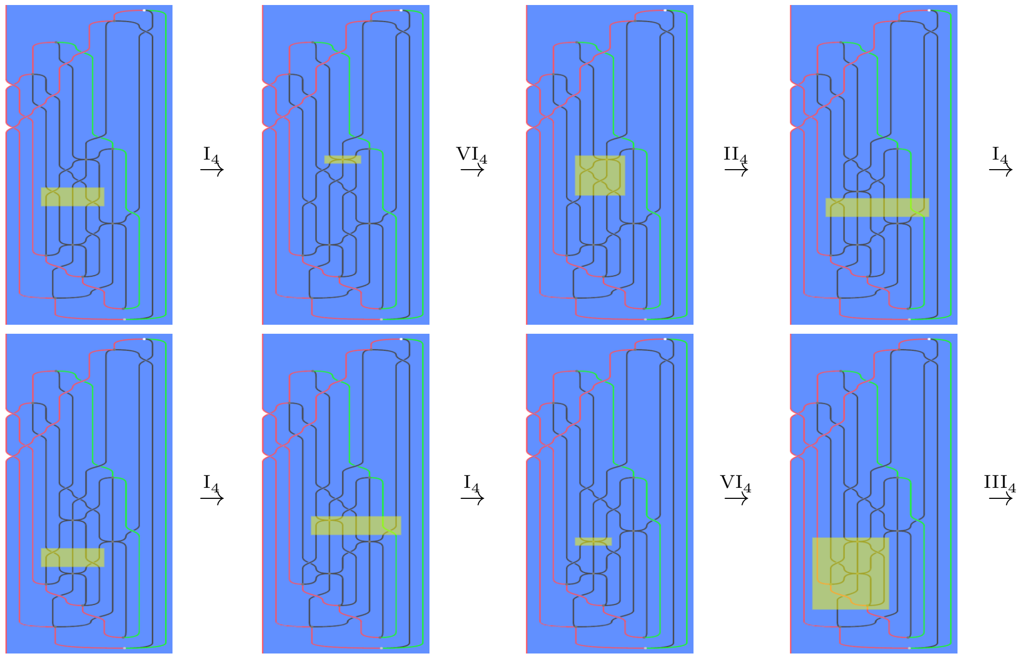
This gets turned into:



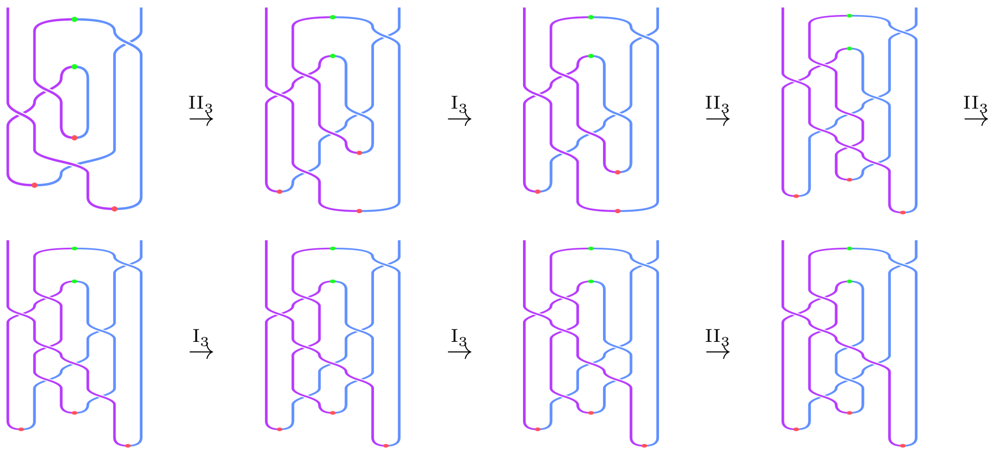
This is an instance of an interchanger of type III, as described in Definition 3.2.12. Note how the 4-cell of type II_3 gets executed right away in the first sequence and then, in the second sequence, gets executed at the end after being pulled-through the purple wire.
We resume the derivation starting with the target of the cell above in the projected view:



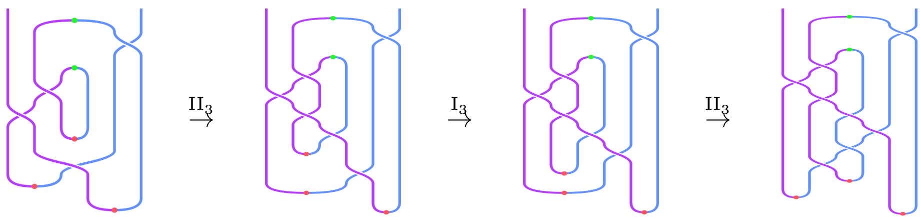

I'_3
I'_3
I'_3
VI_4
II_4
I_4
I_4
I_4
I_4
I'_3
I_4
I_4
I_4
I_4
II_4
I_4
I_4
I'_3
I_4
I_4



To visualise the next step, we illustrate the last slice rendered above in unprojected form, in the neighbourhood of the highlight box indicating the source of the next rewrite:

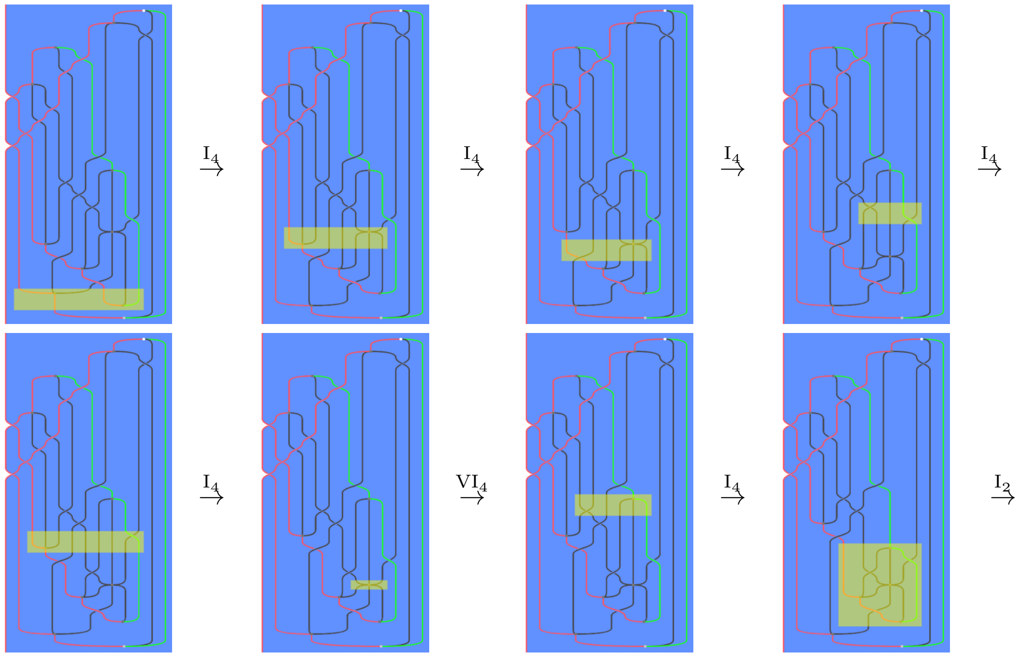


This gets turned into:

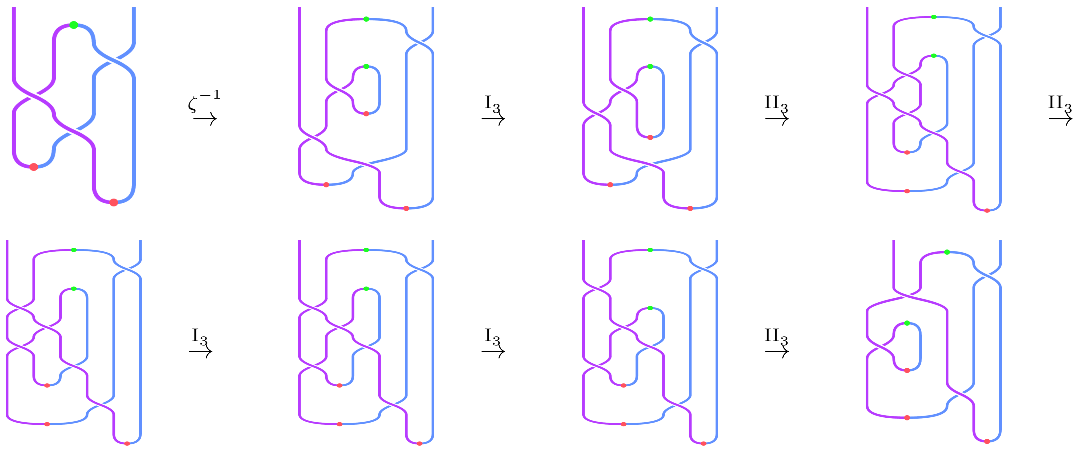


This is an instance of an interchanger of type III, as described in Definition 3.2.12. Note how the 4-cell of type II_3, where the red node is pulled-through the blue wire, gets executed right away in the first sequence and then, in the second sequence, gets executed at the end after being pulled-through the purple wire.

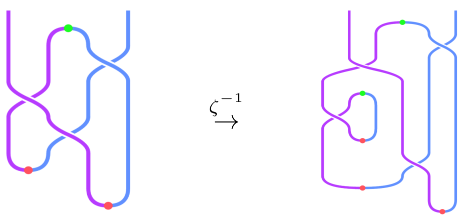
We resume the derivation starting with the target of the cell above in the projected view:



To visualise the next step, we illustrate the last slice rendered above in unprojected form, in the neighbourhood of the highlight box indicating the source of the next rewrite:

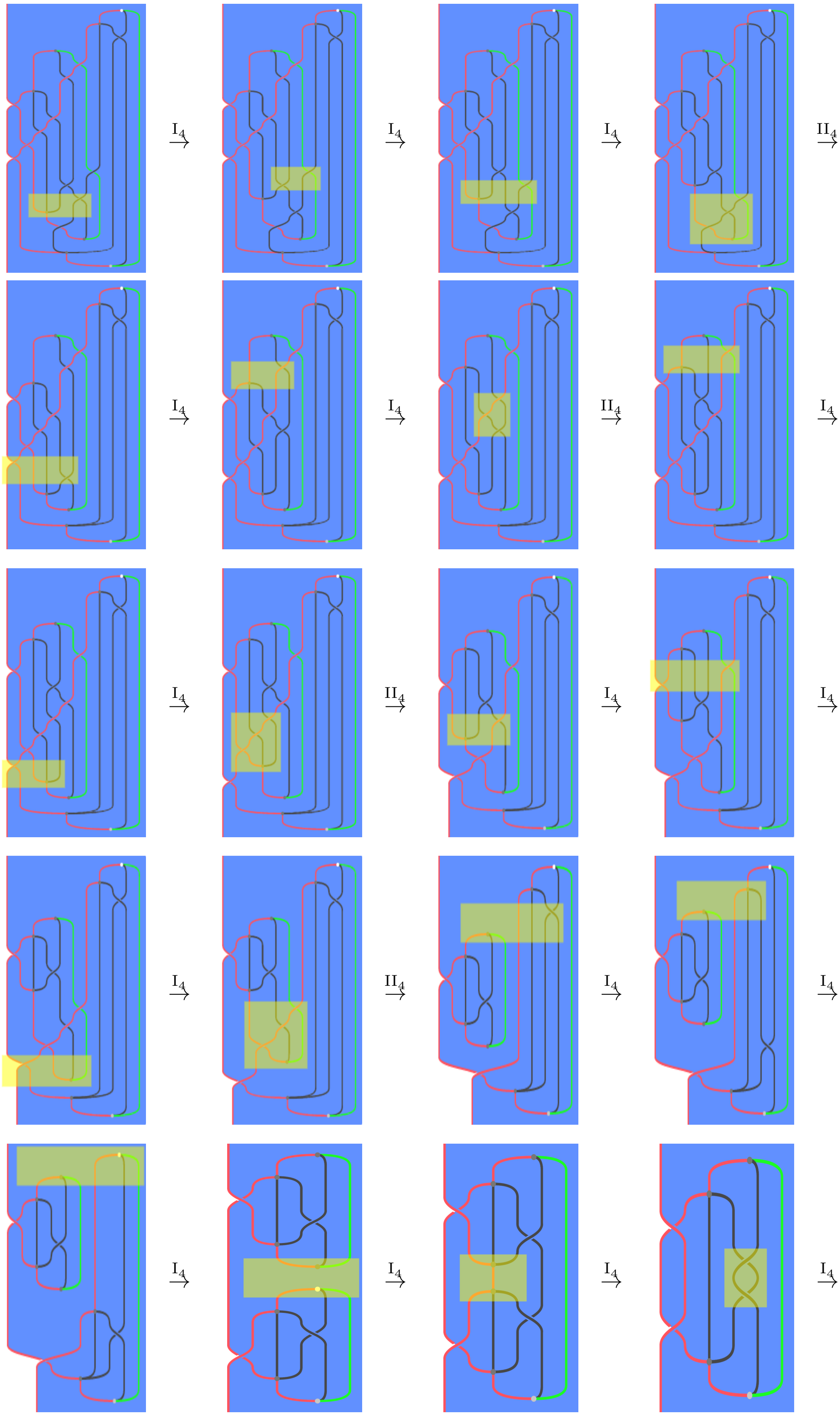


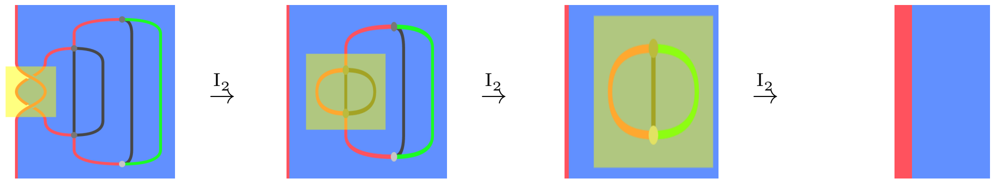
This gets turned into:



This is an instance of an interchanger of type III, as described in Definition 3.2.12. Note how the 4-cell ζ^{-1} gets executed right away in the first sequence and then, in the second sequence gets executed after being pulled-through the purple wire.

We resume the derivation starting with the target of the cell above in the projected view:





By this derivation, the first butterfly equation is satisfied. The other equation follows in a similar manner, hence the adjunction witnessed by objects A, B, 1-morphisms $A \xrightarrow{f} B$, $B \xrightarrow{G} A$, 2-morphisms $f \circ g \xrightarrow{\alpha} \mathrm{id}_A$, $\mathrm{id}_A \xrightarrow{\alpha} f \circ g$ and invertible 3-morphisms Φ', Ψ satisfied the butterfly equations, as required. □

Chapter 6

Complementarity in Higher Quantum Theory

Categorical quantum mechanics is the study of quantum phenomena using methods of category theory that emphasise compositionality. This research program was initiated by Samson Abramsky and Bob Coecke in 2004 [2]. Ever since the program's inception, increasingly sophisticated categorical structures have been employed to express ever broader segments of quantum theory.

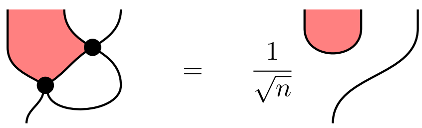
The key milestones achieved are:

- Association of quantum information processing with morphisms in **Hilb**, the category of finitie dimensional Hilbert spaces [2].
- Axiomatisation of observables via special commutative †-Frobenius algebras (classical structures) and their correspondence to bases of the underlying object in **Hilb** [17].
- Usage of classical structures on an object in **Hilb** to define complementarity and to show equivalence with complementarity of orthonormal bases of the same object in **Hilb** [18].

The theoretical foundations of all these advancements are neatly summarised in the set of lecture notes for the Categorical Quantum Mechanics graduate course given at the Department of Computer Science, University of Oxford [61].

One phenomenon that poses significant difficulties in terms of categorical formalisation is quantum measurement, which is a process that turns quantum information into classical information [63]. The most challenging aspect is the ability to capture all possible results of the measurement within one structure. Early axiomatisation attempts involved classical structures and reasoning using post-selection to choose the measurement outcome that is of interest. In that scenario, a story of a particular experiment with a particular measurement result would be told, instead of reasoning about the entire process of measurement. Eventually, the problem was approached using higher categorical structures employed by Vicary [60].

Higher quantum theory is a 2-categorical formalism for reasoning about the flow of quantum and classical information in quantum systems. The crucial fact utilised by the formalism is that classical information can be encoded in correlations between quantum systems. The key advantage of this setup is that many important structures, such as teleportation, dense coding and complementary observables, can be defined by single equations in a symmetric monoidal 2-category. An example of such an equation describing quantum teleportation is given by:



These equations typically have direct physical interpretations, with the defining equation for a structure following immediately from a careful physical description of its required properties. In this chapter, we show that the formalism can be applied successfully to more sophisticated quantum procedures: measurements in a complementary family of bases, quantum key distribution (QKD), and the Mean King problem. For each scenario, we write down a 2-categorical equation that defines the entire procedure in a precise way.

The fundamental 2-cell operations of the higher quantum theory formalism can be combined to form abstract specifications of various quantum information processing protocols. These specifications are independent of implementation details. A 2-categorical specification can subsequently be interpreted in a particular 2-category, assigning a well-typed choice of 0-cells, 1-cells and 2-cells to the appropriate parts of the specification diagram. This amounts to choosing the model, *i.e.* the physical theory in which the specification is to be implemented. Here, we are interested in quantum theory and the correct symmetric monoidal 2-category in which to interpret our specifications is **2Hilb**.

There is a standard diagrammatic notation for reasoning about symmetric monoidal 2-categories presented further in this section, which is closely related to the graphical calculus for semistrict higher categories discussed in Chapter 3. In fact, there is a tentative connection between semistrict n-categories and the contents of this chapter. As predicted by the periodic table of higher categories, a degenerate weak 6-category with the bottom four levels trivialised is a symmetric monoidal 2-category. Hence, if the appropriate coherence results could be obtained, a semistrict 6-category defined as a 7-signature supporting certain interchanger types could model the structures discussed in this chapter.

The main contributions of this chapter can be summarised as follows:

- In Definitions 6.3.1, 6.4.2, 6.4.3 and 6.5.1, we give 2-categorical equations whose solutions in **2Hilb** correspond exactly to implementations of a family of complementary observables, BB84 QKD, E91 QKD, and solutions of the Mean King problem respectively.
- In Theorem 6.4.9, we show that the 2-categorical definition for a family of complementary measurements is equivalent to that for QKD. While an equivalence between these notions seems generally expected in the community, we are not able to find an existing proof in the literature.
- In Theorem 6.5.9, we give a graphical proof of correctness of Klappenecker and Rottleer's solution [33] to the Mean King problem. This is of similar complexity as the original proof, but quite different in nature. The graphical proof puts special emphasis on the role played by complementarity.

A significant result on the categorical basis of quantum key distribution was given by Coecke and Perdrix in [19, Proposition 7.4], which demonstrates the correctness of QKD based on a pair of complementary observables. The results presented here go beyond this, as we work with arbitrary families of complementary observables rather than a single pair, and we further show that every implementation of QKD gives rise to a family of complementary observables. The contents of this Chapter are based on a joint paper with Vicary [8], though only the material which this author has contributed significantly towards is included here.

6.1 Basics of higher quantum theory

2Hilb is a symmetric monoidal 2-category of finite dimensional 2-Hilbert spaces that was first defined by Baez [4]. The necessity of employing such a sophisticated categorical structure is justified by the need to rigorously reason about the process of quantum measurement. Since the 1930s and the introduction of the Hilbert spaces formalism by John von Neumann [62], it is known that certain quantum systems can be mathematically described by a finite dimensional Hilbert space. In the categorical context this is an object in the category of finite dimensional Hilbert spaces - **Hilb**. A measurement performed on this system can produce n different outcomes. If we concentrate only on one of the possible results, we are effectively post-selecting the most convenient measurement outcome and limit ourselves in the understanding of the entire system. Instead we could look at the post-measurement situation as n independent copies of the state space, which is interpreted categorically as an object in the category $\mathbf{Hilb}^n$ - the n-fold Cartesian product of n copies of **Hilb**. Therefore the broader categorical structure that describes the entire system prior to and after the measurement must contain both **Hilb** and $\mathbf{Hilb}^n$. This is satisfied by the 2-category **2Hilb** that up to equivalence consists of the categories of the type $\mathbf{Hilb}^n$.

Definition 6.1.1. The symmetric monoidal 2-category **2Hilb** has *objects* given by natural numbers, *1-morphisms* given by matrices of finite-dimensional Hilbert spaces, and *2-morphisms* given by matrices of linear maps. Details of the compositional structure of **2Hilb** are available in the references given [4,60].

This gives the formal categorical semantics that forms the primary model of our abstract syntax, introduced in the next section.

6.1.1 The topological formalism

The basic 2-categorical structures on which the theory is built have simple graphical representations [60], thanks to the graphical notation for monoidal 2-categories. Its use is again consistent with the overarching aim of this thesis, which is to seek the right level of abstraction to reason about category theory. This graphical formalism involves surfaces, lines and vertices. Their basic interpretation is as follows:

Category theory	**Geometry**	**Interpretation**
Objects	Surfaces	Classical information
1-Morphisms	Lines	Quantum systems
2-Morphisms	Vertices	Physical operations

Composite diagrams involving many vertices are interpreted as a series of actions that take place over time, with time flowing from bottom to top of the picture. In the graphical calculus, as expected, composition of 1-morphisms is given by horizontal juxtaposition, and composition of 2-morphisms by vertical juxtaposition. The tensor product is given by overlaying regions one above the other, perpendicular to the plane of the page and the tensor unit is expressed by an unlabelled, empty region. This is reminiscent of the graphical notation for 3-categories and the consequence of the fact that a monoidal 2-category is a weak 3-category with one object, which here plays the role of the tensor unit.

As the intended application is description of quantum mechanical phenomena, we require the ability to take the formal adjoint of 2-cells, represented graphically by flipping a diagram about the horizontal axis.

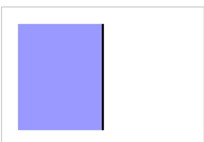
Definition 6.1.2. A *dagger 2-category* is a 2-category equipped with an involutive operation † on 2-cells, such that for all $\mu : F \Rightarrow G$ we have $\mu^\dagger : G \Rightarrow F$, which is functorial and compatible with the rest of the monoidal 2-category structure.

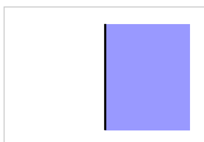
Definition 6.1.3. A 2-cell μ is *unitary* when $\mu \circ \mu^\dagger = \mathrm{id}$ and $\mu^\dagger \circ \mu = \mathrm{id}$.

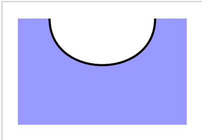
The theory is built on the foundation of a small number of graphical components. They give the formal syntax for our theory for which, as stated before, the semantics is given by the symmetric monoidal 2-category **2Hilb**. We list these components here, and provide their physical interpretations, which are motivated in detail in [60].

Quantum system

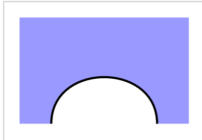

Classical system (6.1)


Right-hand boundary of classical system

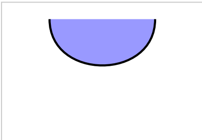

Left-hand boundary of classical system (6.2)



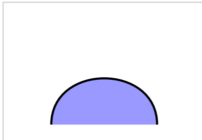
Copy classical information



Compare classical information (6.3)

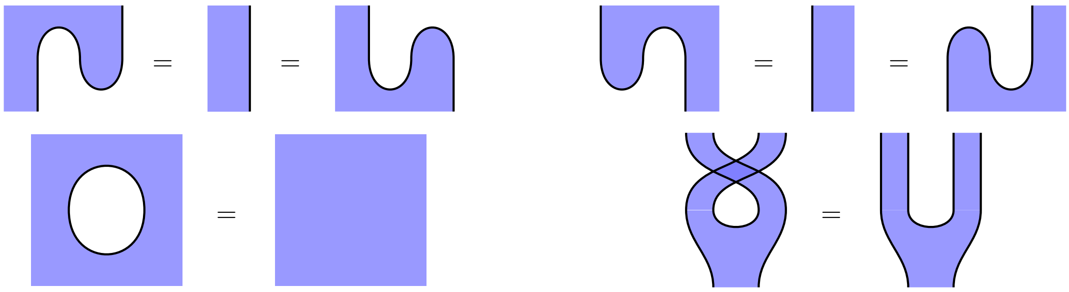


Create uniform classical information



Delete classical information (6.4)

These basic building blocks are required to satisfy a set of axioms, which are captured by saying that the boundary of a region could be smoothly deformed, and that holes in the surfaces can be eliminated:



(6.5)

(6.6)

Any rotation or variants resulting from flipping these equations along the x- or y-axis also hold. The net effect of these axioms is that any two connected networks of copying, comparison, creation and deletion operations, with the same number of inputs and the same number of outputs, will be equal. It follows that every such region carries the structure of a commutative dagger-Frobenius algebra in a canonical way, which is a justification for the association of surfaces with classical information. Note that the symmetric monoidal 2-category structure is used crucially in the last equation here, allowing one region jump into a higher dimension and to pass through another region. We summarise all of the above in the following formal statement:

Definition 6.1.4. In a symmetric monoidal 2-category, an object has a *topological boundary* if it is equipped with the data (6.2)–(6.4) satisfying equations (6.5)–(6.6).

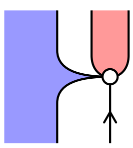
In the reminder of this chapter, we assume that we are working with dagger 2-categories whose objects are equipped with topological boundaries.

In this formalism, an importance is given to correlations between quantum and classical systems, as present in the boundaries of classical systems defined above. In principle, the action of creating these correlations can always be reversed. If we treat quantum measurement as the process of development of correlations between the quantum system being measured and the environment, then this allows us to reverse the measurement procedure. In practice, that would be incredibly difficult to do. Consider, for instance, a correlation established with a randomly passing particle travelling with velocity close to the speed of light. But here, in the theoretical setting, we assume this is possible. For a more in depth explanation of this concept, see [60].

6.1.2 Controlled operations

A pivotal role in the formalism is played by the concept of a *controlled family of measurements* which we define here in a new way. This has the following definition in the graphical language:

Definition 6.1.5. A *controlled family of measurements* is a unitary 2-cell of the following type:



(6.7)

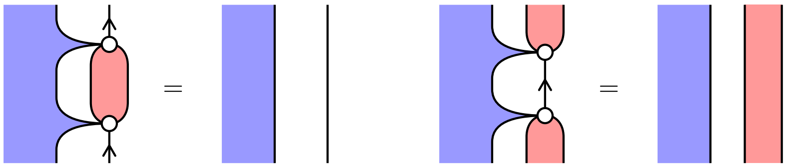
The left-hand pool of classical information represents classical data that determines the basis of the measurement. Throughout this chapter, we always indicate this by colouring the region blue. The line at the bottom-right of the diagram represents the quantum system to be measured. The upper-right pool of classical information represents the classical result of the measurement, which we always draw in red.

We interpret these measurements to be non-degenerate and projective. The motivation for the definition above is made clear by analysing its models in **2Hilb**.

Lemma 6.1.6. *In **2Hilb**, a controlled family of measurements corresponds precisely to a Hilbert space equipped with a list of orthonormal bases.*

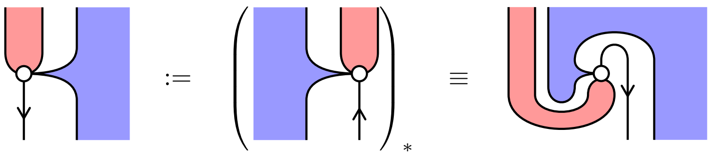
Proof. Let ζ be a 2-cell of type (6.7) in **2Hilb**, additionally let n be the dimension of the blue object, and m the dimension of the red object. Then, ζ constitutes a list of length n, whose entries are m-by-m matrices [60]. For ζ to be unitary means exactly that each m-by-m matrix is unitary, hence we have a list of n unitary operators. However, the red region comes equipped with a canonical commutative dagger-Frobenius algebra structure, and hence a canonical orthonormal basis. Writing the unitaries in terms of this basis, we obtain that the data of ζ is canonically equivalent to a list of n orthonormal bases for the incoming m-dimensional Hilbert space. □

The unitarity property of (6.7) takes the following graphical form:


(6.8)

This gives the impression that the process of quantum measurement can be reversed. This is naturally not the case, as in practice, the red pool of classical information would enter into uncontrolled interaction with the environment and get copied arbitrarily many times. Then, even if one copy of classical information is used to prepare a quantum state, all the other copies remain.

Definition 6.1.7 (Conjugate measurement bases)**.** Following the standard conventions, a controlled measurement with respect to the *conjugate* set of bases is represented by mirroring the diagram about the vertical axis:


(6.9)

Here we decompose the conjugation operation into a composition of adjoint and transpose operations. The blue classical data controlling the choice of basis is now naturally on the right-hand side.

However, we may want to change the side of the classical data controlling the choice of basis *without* passing to the conjugate set of bases. To do this, we use the symmetric monoidal 2-category structure to directly move the blue classical region to the other side. In order to distinguish this from the conjugate controlled measurement (6.9), we represent it as a black vertex.

Definition 6.1.8 (Control from the other side)**.** We use a black vertex to indicate control of the measurement and encoding vertices from the other side:


(6.10)

From this point, arrows indicating dual objects are omitted to increase readability of the pictures.

6.1.3 Projectors

In the course of performing a quantum protocol, we often need to refer to specific values of classical data, such as a the choice of a specific basis in which qubits are prepared in quantum teleportation. For that reason, we need to be able to constrain the value held by a pool of classical data, which we achieve by introducing the notion of a projector.

Definition 6.1.9. Given an object $\mathbf{C} \equiv \mathbf{Hilb}^n$ in **2Hilb**, a *classical data projector* is an element of the canonical n-element basis for the vector space $\mathrm{Hom}_{\mathbf{2Hilb}}(\mathrm{id}_{\mathbf{C}}, \mathrm{id}_{\mathbf{C}})$.

These projectors act to constrain the classical data stored in a region to a particular value. We write them as floating labels that decorate our regions. The following lemma establishes some of their key properties.

Lemma 6.1.10 (Properties of classical data projectors)**.** *For diagrams in **2Hilb**, we can use classical data projectors to decompose the identity, and two adjacent projectors annihilate unless they are identical:*

(6.11)

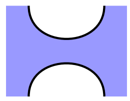
The projectors can move freely around within regions, much like scalars in the theory of monoidal categories. Furthermore, labelled regions can be connected and disconnected arbitrarily:

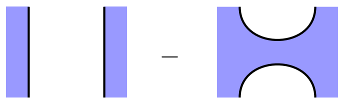
(6.12)

Proof. Straightforward, but omitted for reasons of space. □

We can also define a different type of projector, which constrains the values of two separate regions of classical data to be the same, or to be different. Since we only want to restrict the values of classical information, these projectors are only applied to regions that are coloured blue in our notation. There will always be exactly 2 blue regions in every diagram where we use the projectors, so it will be unambiguous to which regions they 'attach'.

Definition 6.1.11 (Same-value and different-value projectors)**.** In a symmetric monoidal 2-category whose hom-categories are **Ab**-enriched, for an object with topological boundary, the *same-value projector* P_{s} and *different-value projector* P_{d} are defined as follows:

$P_{\mathrm{s}} :=$  (6.13)

$P_{\mathrm{d}} :=$ 

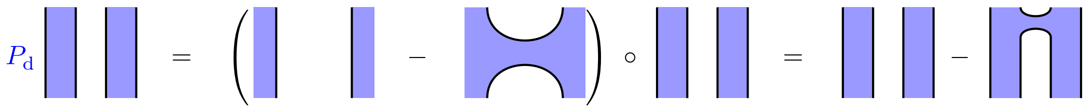
(6.14)

The 2-categorical equations for quantum theory are interpreted in **2Hilb**, in which hom-categories are indeed **Ab**-enriched. There are several properties that we would expect projectors to have, they are summarised as follows:

Lemma 6.1.12. *The projectors* P_{s} *and* P_{d} *satisfy* ${P_{\mathrm{s}}}^2 = P_{\mathrm{s}}$*,* ${P_{\mathrm{d}}}^2 = P_{\mathrm{d}}$*,* $P_{\mathrm{s}} \circ P_{\mathrm{d}} = P_{\mathrm{d}} \circ P_{\mathrm{s}} = 0$ *and* $P_{\mathrm{s}} + P_{\mathrm{d}} = \mathrm{id}$.

Proof. Straightforward graphical proof, omitted for reasons of space. □

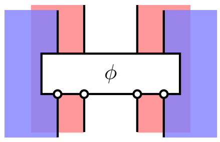
In the reminder of this chapter, we assume that we are working in an **Ab**-enriched 2-category, and so these projectors are well-defined. The tensor product, vertical and horizontal composition in the 2-category all distribute over the additive structure introduced by these projectors. Distributivity of 2-cell composition with respect to addition is illustrated by the following. Note, that we can apply the projectors P_s , P_d whenever the appropriate regions have any open boundary:



(6.15)

6.1.4 Attaching controlled phases

Definition 6.1.13. A *controlled phase* ϕ is an unitary endomorphism of a family of boundaries:



(6.16)

The white nodes decorating the 2-cell ϕ indicate the attachment points to the respective boundaries.

In **2Hilb**, such a structure gives a controlled phase, *i.e.* a family of unit complex numbers, indexed by the values of the classical information of the regions to which the phase is connected. The result of such a controlled phase is to render the overall wavefunction of the system entangled, without introducing any classical statistical correlation between local measurement results [60].

6.2 Complementarity

The concept of complementarity first earned prominence in the 1920s in the aftermath of the great advances in building the mathematical formalism underpinning quantum theory. After Werner Heisenberg published his matrix mechanics and devised the famous uncertainty principle, Niels Bohr's Copenhagen interpretation of quantum theory fully took shape. According to Bohr, quantum states are not states of physical reality, but a probabilistic mixture of all possible measurement outcomes. Inherently tied to this interpretation is the concept of complementarity. For Bohr it served as the philosophical principle explaining the bizarre nature of quantum states. Different classical properties could be joined into complementary pairings: position and momentum, energy and time, particle and wave nature of light, such that no experiment could ever simultaneously reveal the precise value of both with arbitrary precision. In fact, Heisenberg's uncertainty principle puts a bound on the degree of measurement precision in this situation. If we measure one of the complementary properties with perfect accuracy, this implies that we have no knowledge whatsoever about the value of the other property. In this way, the principle of complementarity governs how classical properties interact and combine to produce quantum behaviours.

As is often the case in using quantum theory for the purposes of performing computation, even though at first glance the inability to measure two complementary properties (observables) at the same time may seem to be a limitation, it could be utilised to our advantage and treated instead as a *feature*. In the Hilbert spaces formalism, measuring two observables means that we measure the system with respect to two different bases for the same Hilbert space $\mathcal{H}$ which correspond to these observables. A single pair of non-degenerate measurements is complementary if a standard condition in quantum information, sometimes also known as *unbiasedness* holds.

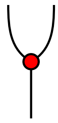
Definition 6.2.1. Two bases $\{|a_i\rangle\}$, $\{|b_j\rangle\}$ of a finite-dimensional Hilbert space $\mathcal{H}$ are *complementary*, or *unbiased*, when for all i, j we have:

$$|\langle a_i|b_j\rangle|^2 = \frac{1}{\dim(\mathcal{H})}$$

The physical interpretation is that if we prepare a quantum state as the eigenstate of one of the two complementary bases and measure it in the other basis, all possible outcomes are equally likely, *i.e.* the uncertainty about the measurement outcome is maximal. The first characterisation of this property in terms of monoidal categories was given by Coecke and Duncan [18], and a 2-categorical definition was given in [60]. Here we provide a short exposition of treatment of complementarity in both formalisms. For monoidal categories, we use the standard graphical notation [50].

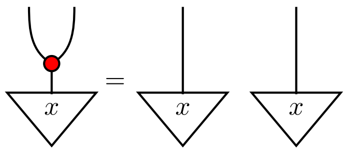
Another way of looking at observables in **Hilb** is through a copying-deleting pair of morphisms:

Definition 6.2.2. In **Hilb**, a *classical structure* on an object A is a copying-deleting pair of morphisms: $A \xrightarrow{\delta} A \otimes A$ and $A \xrightarrow{\epsilon} I$ defining a commutative †-Frobenius Algebra on A. Graphically this is presented as:

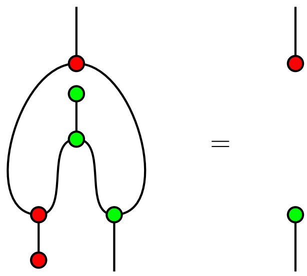
$\delta :=$  Copy $\qquad \epsilon :=$  Delete

These pairs also exactly correspond to orthonormal bases of the underlying Hilbert space [17].We use different colours to indicate different classical structures.

Definition 6.2.3. A state $I \xrightarrow{x} A$ of a classical structure (A, δ, ϵ) is *copyable* when $(x \otimes x) \circ {\rho_I}^{-1} = \delta \circ x$, where ${\rho_I}^{-1}$ is the right unitor isomorphism:



Definition 6.2.4. In a dagger symmetric monoidal category two classical structures $(A, \delta_x, \epsilon_x)$ (indicated by green), $(A, \delta_x, \epsilon_x)$ (indicated by red) are *complementary* if the following condition holds:

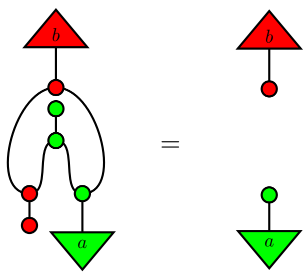
$\dim(A)$  (6.17)

This is the co-called Coecke-Duncan condition for complementarity [18].

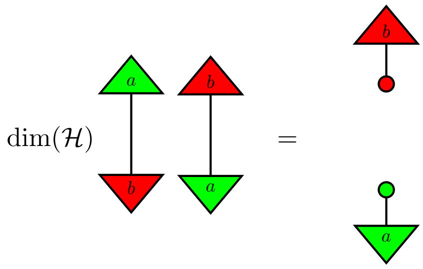
Let $\mathcal{H}$ be a Hilbert space seen as an object in **Hilb**. Then, we say that two orthonormal bases of H are complementary if classical structures on the object $\mathcal{H}$ associated with them are complementary in **Hilb** in the sense of the definition above.

Lemma 6.2.5. *A pair of classical structures in **Hilb** satisfies the Coecke-Duncan condition if and only if the bases that they correspond to are complementary in the sense of Definition 6.2.1.*

Proof. Assume that the Coecke-Duncan condition holds, then the following is equation is true for all copyable states $|a\rangle$ and $|b\rangle$ of the respective bases:

$\dim(\mathcal{H})$  (6.18)

After simplification, using the fact that $|a\rangle$ and $|b\rangle$ are copyable for their bases, the equation takes the following shape:



(6.19)

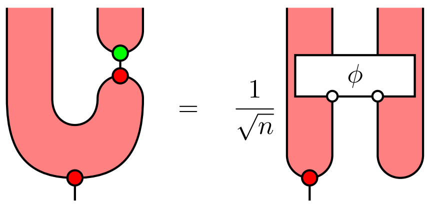
This is equivalent to saying that:

$$\dim(\mathcal{H})|\langle a|b\rangle|^2 = 1 \tag{6.20}$$

Since this has to hold for any pair a, b of elements in the respective bases, we obtain equivalence to the complementarity condition outlined in Definition 6.2.1, as required. □

In the 2-categorical framework we may characterise complementarity through usage of classical information. Following Vicary's presentation [60], successive measurements of the same system in complementary bases produce results that are uncorrelated. This could be expressed by the following diagram where green and red are used to indicate different bases:

Definition 6.2.6. Two measurement bases are *complementary* if there exists some unitary 2-cell ϕ satisfying the following equation:



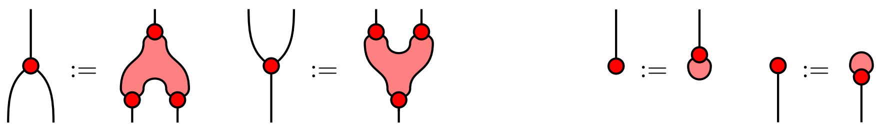
(6.21)

The global phase ϕ depends on each measurement outcome (this is signified by white dots denoting points of attachment) and up to its application the resulting quantum state factorises. Uniform creation of classical data is an isometry, so a normalisation constant is included for this to hold.

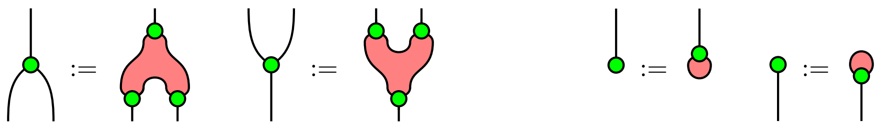
As in the monoidal category formalism, here we could also use classical structures to describe complementarity:

Lemma 6.2.7. *A pair of complementary bases give rise to a pair of commutative †-Frobenius algebras (classical structures) on the symmetric monoidal category of scalars of the symmetric monoidal 2-category* ***Hilb****.*

Proof. The copying-deleting pairs are given by the following:



(6.22)



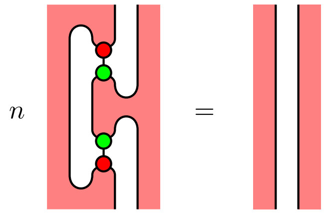
(6.23)

□

There are also two alternative, equivalent formulations of the complementarity condition that concentrate on different aspects of the phenomenon. These are as follows:

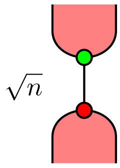
Definition 6.2.8. Two measurement bases are *complementary* if the pair of classical structures that they give rise to satisfied the Coecke-Duncan condition as defined in 6.2.4

Definition 6.2.9. Two measurement bases are *complementary* if the following equation is satisfied:



(6.24)

Equivalently, this is captured by saying that the following 2-cell is horizontally unitary [60]:



(6.25)

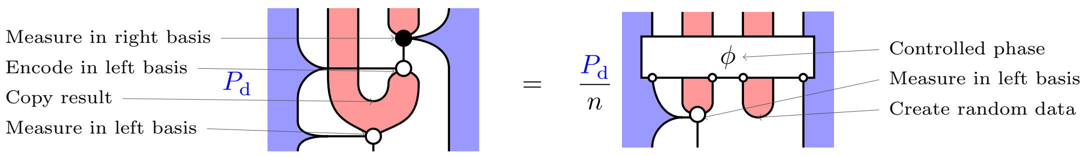
The equivalence of all these conditions for complementarity is shown by simple topology preserving diagram manipulations and 2-cell composition on both sides of the equality [60].

6.3 Complementary families of measurements

In Definition 6.1.5 we introduced a 2-categorical axiomatisation of a controlled family of measurements. In this section, we add the extra requirement that any two distinct measurements in the family are *complementary*. In this case, we say that we have a *complementary family* of measurements. These play an essential role in quantum key distribution and the Mean King problem, which we study in Sections 6.4 and 6.5.

6.3.1 Basic definition

Definition 6.3.1 (Complementary family)**.** A *complementary family of measurements*, or simply a *complementary family*, is an ordinary family of measurements as given in Definition 6.1.5, such that there exists some unitary 2-cell ϕ satisfying the following equation:



(6.26)

The black measurement vertex is as defined in 6.1.8. The pool of classical information controlling the measurement choice gets attenuated, so that the key features of the diagram are not obstructed. The definition has an immediate physical motivation. On the left-hand side, a quantum system is first measured in some particular basis depending on the value of classical information, then the result is copied. One of the copies subsequently gets re-encoded back into a quantum state using the same choice of basis. This quantum state then undergoes another measurement procedure (represented in the picture by the black vertex). The measurement is performed in another basis guaranteed to be different from the first by the presence of the projector P_d. The interpretation of the right-hand side of the equation is that this entire procedure must be equivalent to doing the original measurement with respect to the original basis, but then choosing the second measurement result uniformly at random, up to the application of some overall phase that allows the wavefunctions to be entangled without introducing any classical correlation, this is signified by the presence of ϕ.

Correctness of this definition for ordinary quantum theory follows from previous results on the 2-categorical characterisation of complementary measurements.

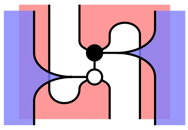
Lemma 6.3.2. *In **2Hilb**, the complementary families are exactly Hilbert spaces equipped with a collection of pairwise-complementary orthonormal bases.*

Proof. First we label the left- and right-hand blue regions on each side of equation (6.26) with distinct projectors a and b respectively. By this, on the right-hand side, we force the controlled measurements to become ordinary measurements in different bases. On the left-hand side, additionally, the left and rightmost attachment points for the phase ϕ disappear. Hence, we obtain the ordinary 2-categorical condition for a complementary pair of orthonormal bases [60]. Conversely, suppose we have a family of orthonormal bases; then by writing the identity as a sum of projectors using Lemma 6.1.10, equation (6.26) follows. □

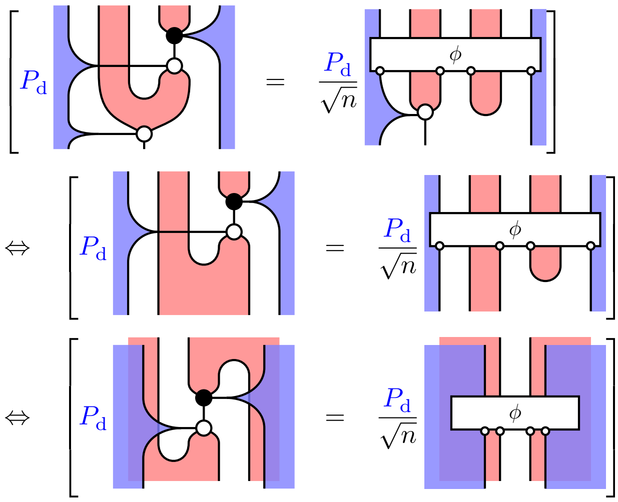
6.3.2 Alternative characterisations

Here we examine alternative characterisations of the complementary family definition.

Lemma 6.3.3 (Complementarity through unitarity)**.** *A controlled family of measurements is complementary if and only if the following 2-cell is unitary on the support of the projector* P_{d}*:*

$\alpha \quad :=$  (6.27)

Proof. We consider the following chain of equivalences:

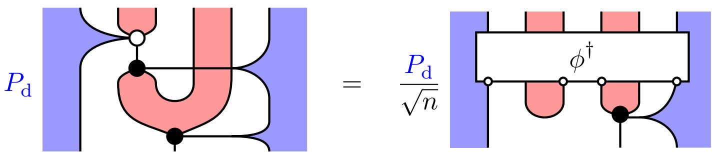


For the first equivalence we compose at the bottom with the inverse of the controlled measurement vertex; for the second we perform a topological manipulation. Since ϕ is an arbitrary unitary 2-cell, it is clear that the last condition is exactly that given in the statement of the lemma.

□

The following alternative formulations of complementarity are used in the analysis of the quantum key distribution protocol and the Mean King problem.

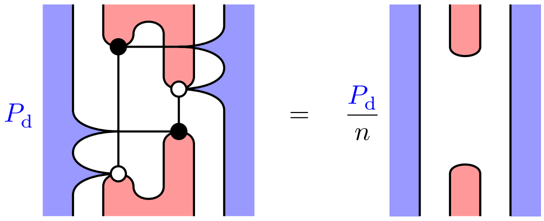
Lemma 6.3.4 (Complementarity condition under horizontal reflection)**.** *A family of controlled measurement operations is complementary if and only if the following equation is satisfied:*



(6.28)

Proof. By Lemma 6.3.3, both α and $\alpha^\dagger$ are unitary on the support of the projector P_{d}. The condition 6.28 can be obtained by elementary 2-cell operations from the unitarity of $\alpha^\dagger$. □

Lemma 6.3.5 (Alternative formulation of complementarity)**.** *A controlled family of measurement operations is complementary if and only if the following condition is satisfied:*



(6.29)

Proof. The result is proved using Lemma 6.3.3 and by performing topological manipulations. □

6.4 Quantum key distribution

In the final two sections of this chapter, we present how the concept of complementarity could be utilised to realise computational tasks unachievable with classical computation. We consider two applications: quantum key distribution (QKD) and a perhaps slightly less well-known, Mean King problem (MKP). In both problems, complementary bases play a crucial role in the success of the protocol. The abstract framework used to describe complementarity in the previous section will now be applied to graphically illustrate these protocols and prove their correctness.

First, in this section we give 2-categorical equations defining two different variants of quantum key distribution. The variants are referred to in the literature as BB84 [11] and E91 [23] with both names deriving from the first letters of their discoverers' surnames and the year when the protocols were first proposed. In Theorem 6.4.4 we show that these two forms are topologically equivalent. The main result of this section is Theorem 6.4.9, in which we demonstrate that these quantum key distribution equations are equivalent to Definition 6.3.1 of a complementary family of measurements.

A quantum protocol consists of two parts, the set of instructions and the desired behaviour. The set of instructions is an ordered list of operations to perform in order to achieve the desired behaviour, otherwise referred to as the goal of the protocol. Throughout the remaining part of this chapter we implicitly use the following definition to reason about correctness of quantum protocols [28]:

Definition 6.4.1. We say that a quantum protocol is *correct* (or valid) if its set of instructions implies the desired behaviour.

In our 2-categorical diagrammatic specifications, the set of instructions is expressed on the left-hand side of the equation and the desired behaviour on the right-hand side.

6.4.1 Quantum key distribution

A major development in the theory of cryptography is due to Shannon [51], who showed that a secret message may be transmitted with perfect secrecy if both parties share identical one-time pads that are used to encode and decode the message. Using this observation, the problem of secure transmission of a secret message is reduced to the problem of sharing a one-time pad, or a *key* between the parties. This is the main motivation for developing various key distribution schemes. The task of sharing a key is less challenging than the task of sharing a secret message because the one-time pad is allowed to be random. Additionally, if we were able to detect the presence of eavesdroppers in the process of transmitting the key, the communication could be terminated immediately without divuldging any section of the secret message to third parties. This is where we could utilise quantum complementarity as a resource.

In the following presentation we concentrate on distributing one bit of the secret key. To obtain keys of arbitrary length, the same procedure needs to be performed repeatedly until the desired number of key bits has been shared.

The two most well-known protocols realising quantum key distributions are BB84 [11] and E91 [23].. Here, we provide a short outline of both protocols. The set of instructions for distributing one key bit via BB84 is as follows:

(1) Alice randomly chooses either 0 or 1. Then, she randomly picks one of the two pre-agreed complementary bases and prepares a qubit in that basis, this encodes her classical information. Finally, she sends the qubit to Bob.

(2) After receiving the qubit from Alice, Bob also randomly picks one of the two complementary bases and measures the qubit in that basis.

(3) Both Alice and Bob share their basis information over a classical channel. If they chose different bases, they discard the bit. If the basis information matches, they establish Alice's random bit as the key bit.

One observation to be made is that should Bob choose the correct basis (the same as Alice), his measurement reveals Alice's random bit. If however he chooses wrongly, his measurement result is going to be completely random due to complementarity of the bases used.

E91 employs a pair of entangled particles as a resource that allows both Alice and Bob to randomly choose a basis in which to measure their qubits in. The set of instructions is as follows:

(1) Alice and Bob prepare an entangled Bell state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ and each take one of its qubits.

(2) Alice and Bob each randomly pick one of the two pre-agreed complementary bases and each measures their qubit in their chosen basis.

(3) They share their basis information through a classical channel. Should they pick different bases, they discard the measurement results and start from scratch. Otherwise, they successfully shared one bit of the secret key.

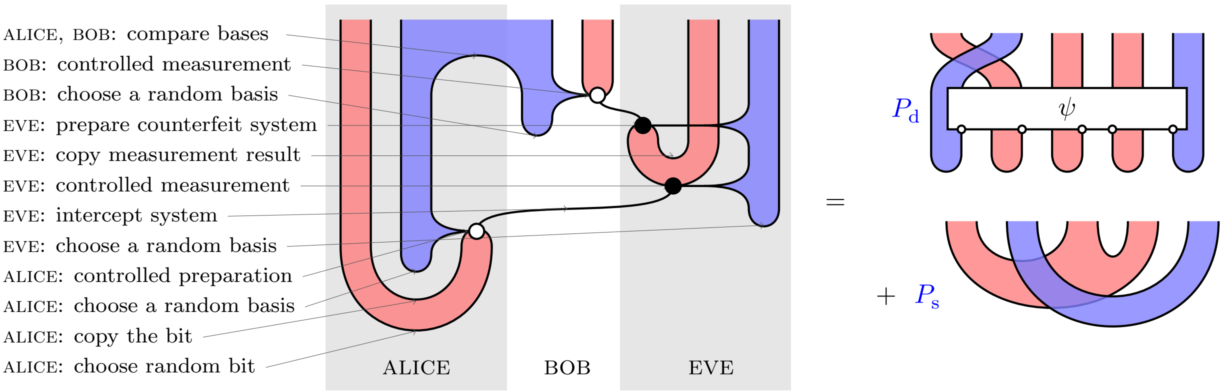
In the presence of a malicious eavesdropper Eve we assume that she has access to all quantum and classical channels of communication between Alice and Bob, as well as that she has knowledge about all the prearrangements made by the parties, such as the set of bases to be used.

Eve's objective is to extract information about Alice's basis with the additional requirement that she disrupts the channel between Bob and Alice as little as possible. If Alice uses a random basis to encode her random bit, the best Eve can do is to take a guess herself and pick the right basis 1/2 of the time. In this scenario she does not disturb the channel and is not detected. However in the long run, she will pick the wrong basis 1/2 of the time. If Alice uses complementary bases, Eve's incorrect choice of basis results in her obtaining a random result and to make matters worse, there is a 1/2 chance (1/4 of all cases) that she disturbs the quantum state in a detectable way. Eve's interference is detected when Bob obtains a measurement result different that Alice's bit, despite the fact that they used the same measurement bases.

The scenario that interests us is when Alice and Bob use complementarity to prevent Eve from obtaining information about their communication. This happens when both Alice and Bob pick the same basis and Eve randomly picks a different basis.

6.4.2 Abstract definitions

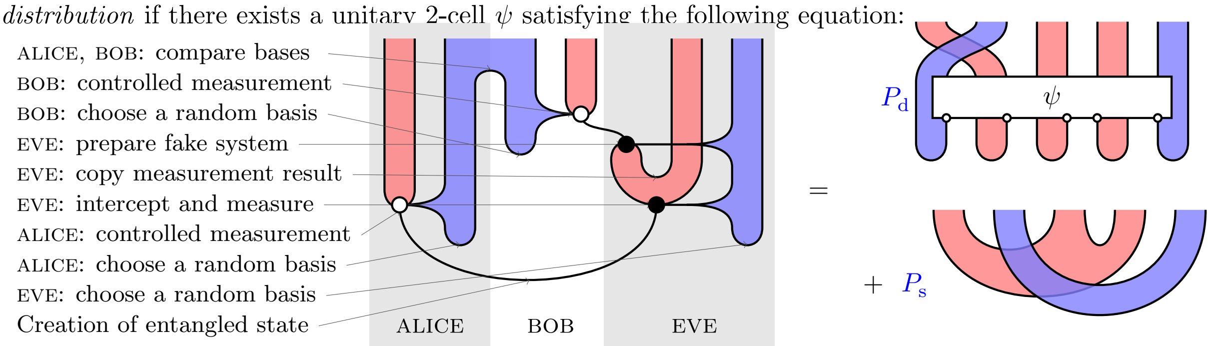
Definition 6.4.2 (BB84 QKD)**.** A controlled family of measurements satisfies *BB84 quantum key distribution* if there exists a unitary 2-cell ψ satisfying the following equation:



Each of the diagrams on the right-hand side corresponds to the desired behaviour depending on whether Eve guessed the basis correctly. If Eve guesses incorrectly, then the P_d term says that both basis choices and all 3 measurement results are classically uncorrelated. If Eve guesses correctly, then the P_s term says that Eve shares Alice and Bob's basis, and that all three share the same classical data.

A different equation can be obtained from consideration of the E91 QKD protocol.

Definition 6.4.3 (E91 QKD)**.** A controlled family of measurements satisfies *E91 quantum key distribution* if there exists a unitary 2-cell ψ satisfying the following equation:



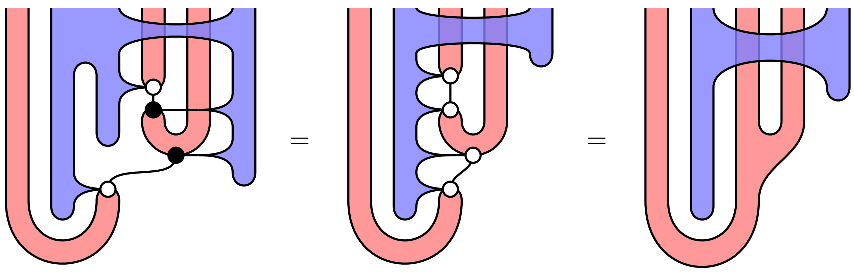
Theorem 6.4.4. *The equations for BB84 and E91 QKD are equivalent.*

Proof. Elementary topological manipulation. ☐

For that reason, in the reminder of this section we only consider the BB84 protocol.

Lemma 6.4.5 (Eve's successful interference)**.** *On the support of projector P_s, the quantum key distribution specification is satisfied for any controlled family of measurements.*

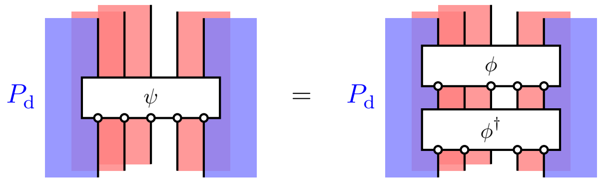
Proof. We investigate this scenario by applying the projector P_s on both sides of the specification. In this case the right-hand side only retains the P_s component, and the left-hand side simplifies as follows:



Vertex colour changes are justified by changing the side from which the operations are controlled in accordance with Definition 6.1.8. By this, we can conclude that after application of P_s the QKD specification becomes a tautology. ☐

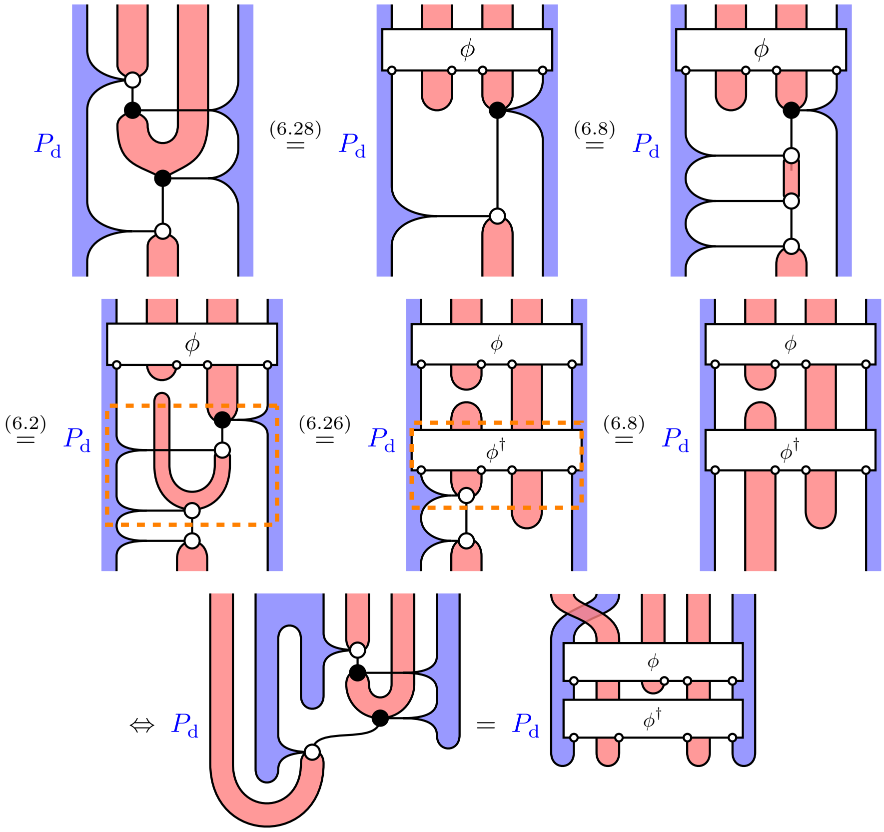
6.4.3 Quantum key distribution from a complementary family

Lemma 6.4.6. *If a controlled family of measurements is complementary, then it satisfies the quantum key distribution specification with:*



(6.30)

Proof. Suppose the controlled complementarity condition 6.3.1 is satisfied. Then we make the following argument:



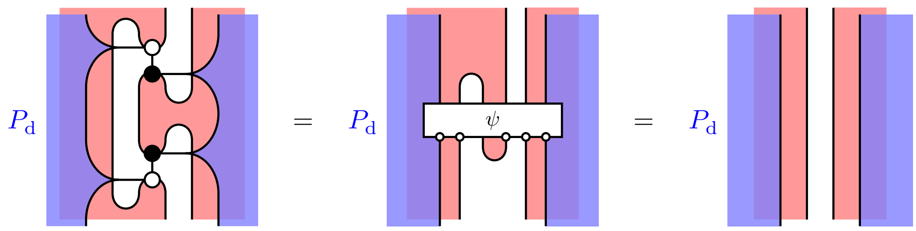
The final equality follows from the first chain of equalities by topological deformation. This final equality is equivalent to the statement of BB84 quantum key distribution as given in Definition 6.4.2, since by Lemma 6.4.5 the P_s component is trivially satisfied.

□

6.4.4 A complementary family from quantum key distribution

Lemma 6.4.7. *If a controlled family of measurements allows quantum key distribution with a phase ψ, then:*

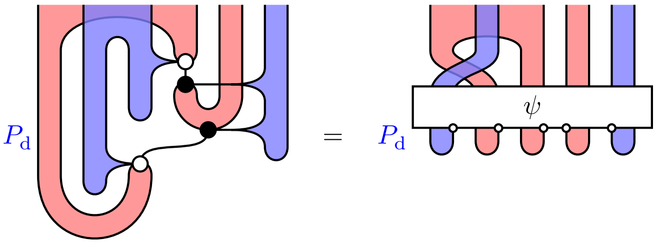
$\alpha^\dagger \circ \alpha$ =



(6.31)

Here α is as defined in the proof of Lemma 6.3.3.

Proof. If Eve picks the wrong basis but does not influence the communication between Alice and Bob, their key information is still the same. We post-select on this scenario by applying a projector P_{d} to pools of classical information corresponding to Alice's, Bob's and Eve's basis information and by applying the comparison operation to pools corresponding to Bob's and Alice's key information:

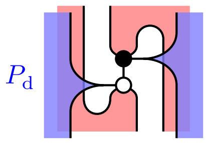


(6.32)

Using topology preserving elementary 2-cell operations, the first equality in 6.31 is obtained. For the second equality, up to application of P_{d} on the outer pools of classical information, the middle 2-cell in equation 6.31 is a unitary, since ψ is a unitary. Also, $\alpha^\dagger \circ \alpha$ is a positive map. The only positive unitary is the identity, hence the result is established. □

Lemma 6.4.8 (Quantum key distribution implies complementarity)**.** *If a controlled family of measurements allows quantum key distribution, then the family is complementary.*

Proof. By Lemma 6.4.7 the following map is unitary:

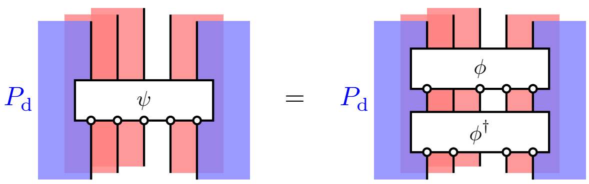


By Lemma 6.3.3, we can conclude that the controlled family of measurements is complementary. □

Theorem 6.4.9. *A controlled family of measurements satisfies quantum key distribution if and only if it is complementary.*

Proof. Immediate by Lemmas 6.4.6 and 6.4.8. □

Lemma 6.4.10. *If a controlled family of measurements allows quantum key distribution with phase ψ, then we can decompose ψ in the following way:*



(6.33)

Proof. Immediate by Lemmas 6.4.6 and 6.4.8. □

6.5 The Mean King problem

The original formulation and solution of the Mean King problem for three complementary observables is due to Vaidman, Aharonov and Albert [3]. However it is often presented in the literature in the amusing form of a fairy tale [33]. It tells the story of an evil king who hates physics and imprisons an innocent quantum physicist Alice, who has to succeed in a quantum theoretical challenge to regain ger freedom. More formally, the Mean King protocol is defined as follows [3, 33]. There are two agents, Alice and the King, who take part in the following procedure.

1. Alice prepares a quantum state of her choice and hands it to the King.
2. The King measures the state in one of n mutually unbiased bases, keeping his choice of basis and the outcome secret, and returns the state to Alice.
3. Alice performs any quantum measurement she wishes.
4. The King reveals his measurement basis to Alice.
5. Using only classical processes, Alice must calculate the King's earlier measurement outcome.

Let us consider what is entailed by the condition that Alice must determine king's measurement result immediately after he reveals his basis. This essentially means that Alice must be ready to reply straight away, regardless of what the king tells her. Hence, even if the King is deceitful and informs Alice that he measured his state in a different basis than he actually did, she still has to have an answer ready. Moreover, this answer must be such that it would have been correct had the king really measured in the basis he told Alice. This implies that the measurement that Alice performs must provide her with a lookup table for the King's measurement result.

From this introduction it might seem like we require Alice to predict the result of measurement of complementary observables. A requirement that would clearly contradict the rules of quantum mechanics and what we know about uncertainty of determining values of complementary properties. But a more careful investigation shows that, since the king is only physically measuring in one of the complementary bases and other measurements do not happen, Alice is only *retrodicting* the result - inferring it from one particular post-selected scenario. As noticed by Metzger, other scenarios have no physical meaning [41]. This is in contrast to being able to *predict* the result of complementary measurements. The possibility of assigning unit probabilities to results of a set of hypothetical measurements in complementary bases has been shown by Aharonov et al. [3].

Now, we consider specific protocols solving the Mean King problem for n mutually unbiased bases the King uses for $n = 2$ and $n = 3$. For $n = 1$, the solution is trivial and by preparation of an eigenstate of the king's basis, Alice is able to predict the measurement outcome.

For $n = 2$, Alice prepares an eigenstate of one of the two complementary bases that the King may use. Once she receives the qubit back, she measures it in the other basis, one she did not use to prepare the qubit. That way, she knows the result of the possible measurement in one basis even before the king's measurement is peeformed. She is able to retrodict the result of the other possible measurement without any further action, as it is exactly the same as the result of her measurement. This is because the King altered the prepared state by his measurement operation.

Alternatively, Alice could use entanglement to complete the King's challenge. Let Alice prepare the Bell state $\varphi = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, send the second qubit to the King and hold on to the first. After the qubit is returned, Alice randomly chooses one of the two basis to measure the first qubit and then measures the second qubit in the other basis. By this, she is certain that at least one of the qubits is measured in the basis used by the King. It is worth noting that these two solutions are in fact topologically equivalent, and by examining the 2-categorical diagrams it is easy to derive one of them from the other. This is similar in nature to the equivalence between different variants of quantum key distribution protocols.

The solution utilising entanglement can be used to perform quantum key distribution as exemplified by Bub and Yoshida [13, 64]. Alice and the King share one bit of the secret key through successful completion of the procedure above - the King's measurement result serving as the key bit. In fact, Yoshida uses exactly the solution presented above, Bub's approach is more complicated and uses the measurement that also solves the problem for $n = 3$.

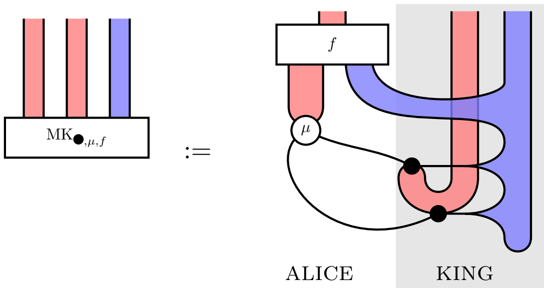
If the King uses three mutually unbiased bases, we must use entanglement. As previously, Alice prepares the Bell state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ and sends the second qubit to the King. He measures the qubit in one of three mutually unbiased bases and returns the qubit to Alice, maintaining information about the result. Alice performs a special bi-partite measurement that we will describe in detail below. After learning the King's basis, Alice applies a classical function (predetermined from the beginning) to find the result in her look up table.

In other words, she should perform some non-degenerate PVM μ on both systems, and prepare a lookup table f that tells her, depending on the King's measurement basis choice and her own measurement result, what King's result was. The key results of this section is a graphical definition of a solution of the Mean King problem, and a graphical proof of the correctness of Klappenecker and Rötteler's solution [33] to the Mean King problem.

6.5.1 Abstract definition

We begin with an abstract definition of the Mean King problem. In categorical quantum mechanics, a *classical function* is defined as a morphism between classical data which satisfies the comonoid homomorphism property, and that regions with topological boundary carry a canonical comonoid structure.

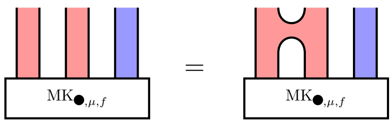
Definition 6.5.1 (Mean King scheme)**.** Given a complementary family of measurements $\bullet$, a bipartite measurement μ, and a classical function f, a *Mean King scheme* $\mathrm{MK}_{\bullet,\mu,f}$ is defined as the following composite:



(6.34)

The black dot denotes measurement in a set of mutually unbiased bases.

Definition 6.5.2 (Mean King solution)**.** A Mean King scheme $\mathrm{MK}_{\bullet,\mu,f}$ *solves the Mean King problem* if the following equation holds:



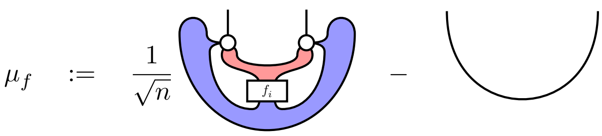
(6.35)

This says exactly that, after carrying out the procedure, Alice and the King carry the same measurement result information, which is indicated by the presence of the projector on the right-hand side of the equation. This requirement is satisfied precisely if the Mean King scheme $\mathrm{MK}_{\bullet,\mu,f}$ is correct.

6.5.2 Solving the Mean King problem

Our solution to the Mean King problem is presented entirely graphically. It is based on a solution due to Klappenecker and Rötteler [33]. Giving this solution graphically is an interesting exercise in the graphical formalism for symmetric monoidal 2-categories, and demonstrates that it is capable of reasoning about sophisticated schemes. Our presentation is comparable in complexity to Klappenecker and Rötteler's. One advantage of our presentation is that it is perhaps clearer in expressing how complementarity is being used. We begin by giving a scheme to construct a bipartite state from a classical function.

Definition 6.5.3. Given a classical function $f_i : n \to m$, and an n-fold controlled family of measurements on $\mathbb{C}^n$, the associated bipartite state $|\mu_f\rangle \in \mathbb{C}^n \otimes \mathbb{C}^n$ is defined as follows:



Definition 6.5.4 (Collisions)**.** Given classical functions $f, g : n \to m$, let $f \diamond g := |\{a | f(a) = g(a)\}|$ be the number of *collisions* between them.

Lemma 6.5.5. *Let $f, g : [n+1] \to [n]$ be functions. Then $n\langle \mu_f | \mu_g \rangle + 1 = f \diamond g$.*

Proof. A straightforward graphical proof is possible, which we omit. □

Lemma 6.5.6. *Given a family of n^2 classical functions $f_i : [n+1] \to [n]$ with $i \neq j \Rightarrow f_i \diamond f_j = 1$, the states $|\mu_{f_i}\rangle$ form an orthonormal basis.*

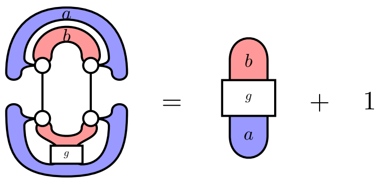
Proof. Rearranging the result of Lemma 6.5.5, we see that $\langle \mu_f | \mu_g \rangle = ((f \diamond g) - 1)/n$, by this argument the conclusion follows. □

Lemma 6.5.7. *For any prime power $n = p^k$, the following structures exist:*

1. *a family of n^2 functions $f_i : [n+1] \to [n]$, such that for $i \neq j$ we have $f_i \diamond f_j = 1$;*
2. *a family of $n+1$ mutually complementary bases.*

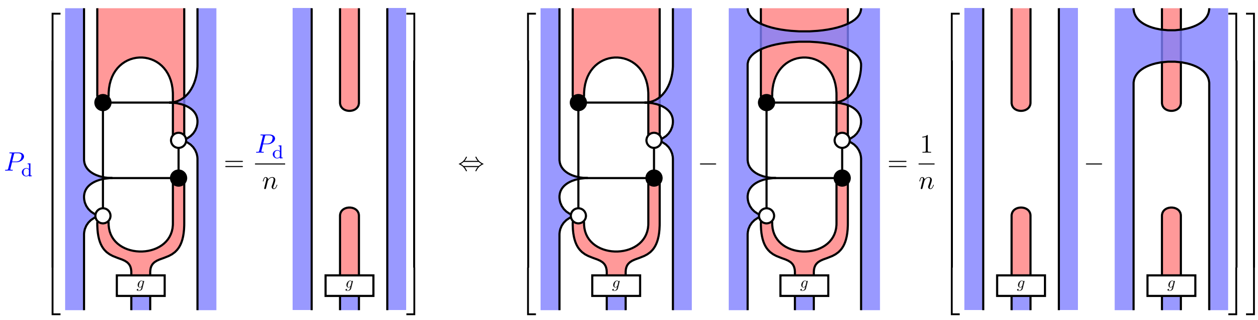
Proof. See [33, Section 2]. □

Lemma 6.5.8. *For a complementary family of controlled measurements and a classical function g, the following holds:*

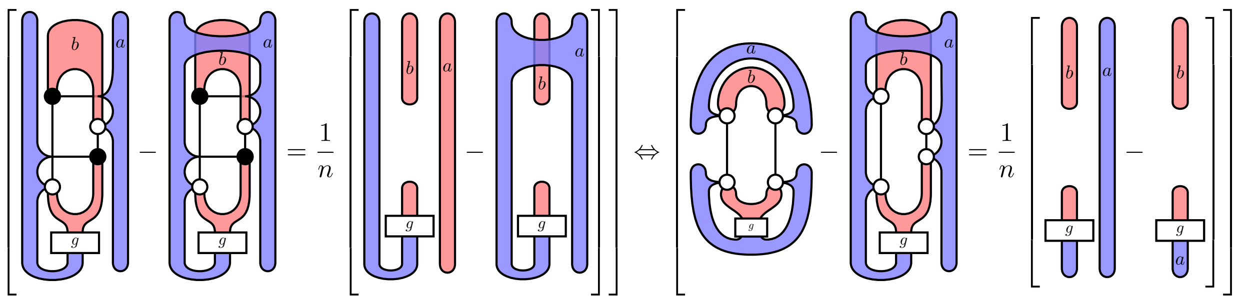


(6.36)

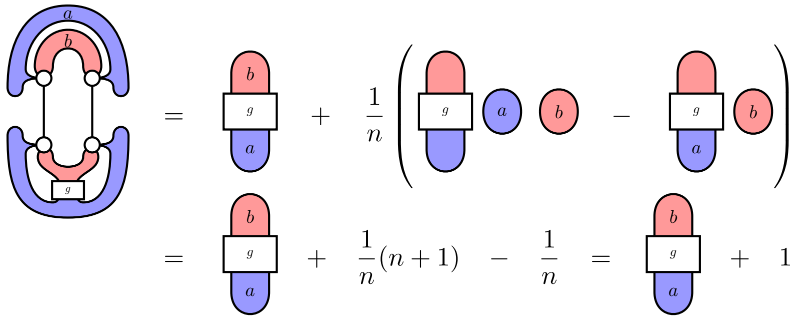
Proof. We use the fact that these controlled measurements form a family of complementary controlled operations. Hence equation (6.29) holds, and we combine this with the classical function g to obtain the left-hand side given below:



The right-hand side is obtained by expanding out the action of the projectors P_d. We next assign specific values a, b to pools of classical information and perform elementary 2-cell operations. We can replace black vertices with white, as long as we switch the side from which the vertex is controlled. Since pools of classical information exhibit topological behaviour, we can reposition them freely.



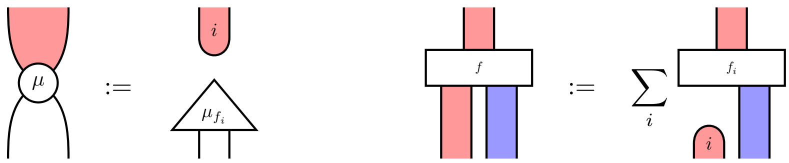
After we cancel out measurement and encoding operations controlled by the same pools of classical information, the equation is simplified to:



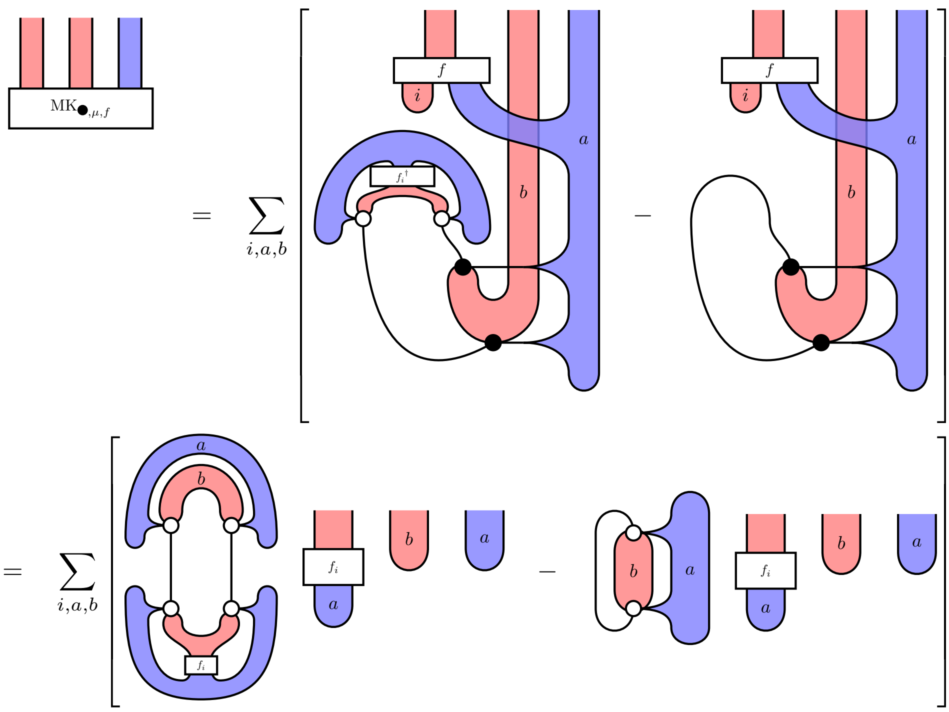
(6.37)

□

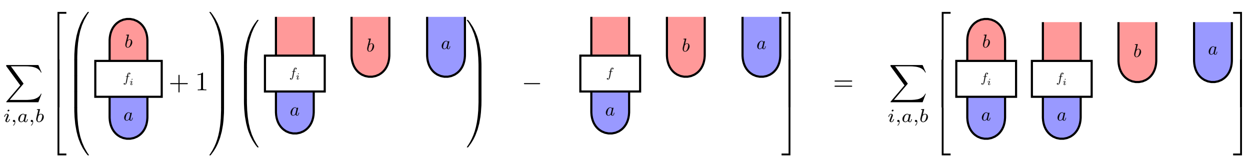
Theorem 6.5.9 (Solution to the Mean King problem)**.** *For a family of functions $f_i : [n+1] \to [n]$ such that $|\mu_{f_i}\rangle$ form a basis, and a family* ● *of $n+1$ complementary bases of $\mathbb{C}^n$, the following assignments give a Mean King solution:*

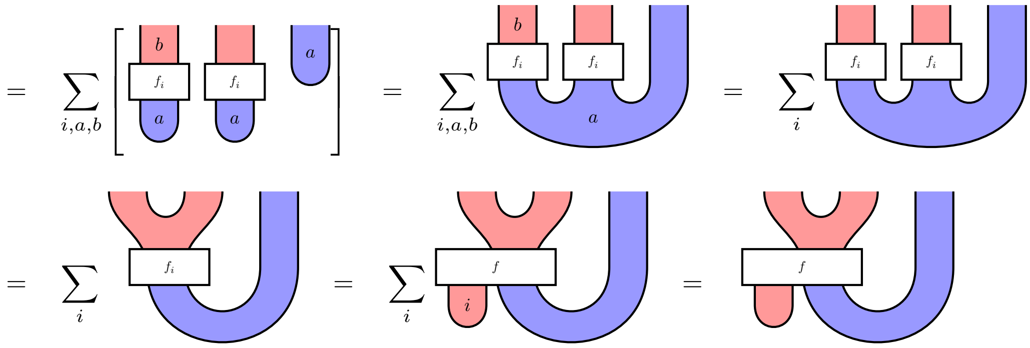


Proof. By Lemma 6.5.7 a suitable family of n^2 functions $f_i : [n+1] \to [n]$ and a complementary family of controlled measurements in $n+1$ bases exist. The latter by defining a controlled operation to pick one of the $n+1$ complementary bases to measure in. For each f_i we define a state μ_{f_i} in accordance with Lemma 6.5.3. By Lemma 6.5.6 states $|\mu_{f_i}\rangle$ form an orthonormal basis μ that we use to solve the problem. The scheme $\text{MK}_{\bullet,\mu,f}$ then simplifies to:



By Lemma 6.5.8, this simplifies as follows:





The final diagram clearly remains unchanged under application of the projector as per Definition 6.5.2, hence the result is established. □

Chapter 7

Conclusion

The main original contributions of this thesis can be summarised as follows:

(1) In Chapter 2, we presented a new theory of generic-position higher-dimensional diagrams, together with their combinatorial description. The theory is based on two mutually-recursive structures: diagrams and signatures, which build on the notion of an n-polygraph endowed with an order structure on n-cells in a diagram. We gave definitions of rewriting and composition of diagrams, and proved correctness properties for these definitions. We also showed results on associativity and distributivity of the latter, allowing signatures to become a foundation for definitions of higher quasistrict n-categories.

(2) In Chapter 3, we used the signature structure to provide a new framework for the definition of quasistrict higher categories requiring considerably fewer axioms than traditional approaches. This is achieved as all associativity and distributivity results are already built-in within the signature structure. It also allowed us to put emphasis on the higher-level coherences which arise from the introduction of the interchange law. We explored these coherences using a graphical formalism that absorbs many low-level details into the notation. As a result, we recovered standard definitions of quasistrict 2- and 3-categories and a new definition of a quasistrict 4-category. We contrasted this approach with attempts due to Crans [20] and Douglas and Henriques [22] and highlighted how our less-strict definition results in simpler, shorter proofs and equivalent expressivity.

(3) In Chapter 4, we discussed how the theoretical framework developed in Chapters 2 and 3 was adapted into a practical proof assistant Globular, which is the tool of its kind. We commented on the design choices made and outlined the main algorithms implemented. We also presented a short demonstration of the tool's capabilities. The chapter is concluded with the list of example proofs formalised with the aid of the tool by members of the community.

(4) In Chapter 5, we proved that an adjunction of 1-morphisms in a quasistrict 4-category gives rise to a coherent adjunction satisfying the butterfly equations, utilising the new definition provided in Chapter 3. As far as we are aware, this is the first time when, an explicit proof carried out in a 4-categorical setting is given in the literature.

(5) Finally, in Chapter 6, we presented an application of the higher categorical formalism to quantum theory. In this framework, we provided a description of a family of complementary bases and gave a completely syntactic proof that a basis satisfies quantum key distribution if and only if it is mutually unbiased. We also gave a logical correctness proof of Klappenecker and Roettler's [33] construction of a solution to the Mean King problem from a family of mutually unbiased bases. All these results were achieved within the framework of symmetric monoidal 2-categories that allows us to fully capture the notion of quantum measurement in an abstract way. A graphical notation is used throughout to explicitly illustrate the flow of both classical and quantum information within the systems considered.

Overall, we showed that finding the right level of abstraction when reasoning about higher categories may give additional insight and put special emphasis on the aspects that are of highest interest in the particular scenario being investigated. We hope that the combination of the three methods—graphical languages, quasistrictness and automated reasoning—will ultimately bring about a deeper understanding of the theory of weak n-categories and its applications to an ever growing family of mathematical disciplines.

7.1 Future work

We conclude with a brief discussion of some possible future directions of research.

7.1.1 Quasistrict categories for $n \geq 5$

The most natural extension of the results of this thesis is further exploration of higher-level singularities that arise from the introduction of the interchange law. The first step towards that, is to investigate all coherences that arise first for a quasistrict 5-category. In most general terms, any two different combinations of interchangers of types I − VI that that have the same effect on a 5-cell α need to be related by a higher-level coherence. As discussed in Chapter 3, many of them are just going to be higher dimensional incarnations of the types already covered. However, new types are also bound to arise.

The most significant difficulty in the exploration of higher-level interchangers is the inability of the human mind to perform visualisations of 5-dimensional geometrical structures. Even though, the techniques described in Chapter 3 and projected images in Globular offer substantial insight, much effort is still required.

There are two other remaining difficulties with regards to describing higher-level singularities for $n \geq 5$. The first is the problem of reliably generating all the possible coherences for the given dimension. Despite the fact that coherences for $n = 5, 6$ could perhaps be classified using trial and error, a more general set of rules is needed if a definition of a quasistrict n-category for an arbitrary n is to be achieved.

One approach discussed is to analyse interactions between different pairs of interchanger types, however this is already not complete for $n = 4$, where type VI_4 does not get generated. Another possibility is to analyse naturality properties for each interchanger, however, again interchangers of type VI cannot be classified using this approach. Additionally, an open question that remains is whether there is an interchanger type whose source and target consists of three or more other interchanger types. All types discussed so far have sources and targets consisting of at most two other interchanger types.

The second problem is to be able to prove that all coherences for the given dimension have indeed been generated. One method to achieve that is to prove equivalence of the created quasistrict n-category to a general weak n-category. However this is yet to be achieved even for $n = 4$.

We hope that the visualisation capabilities of Globular will turn out to be of major assistance in the efforts to further explore the space of higher-level coherences in quasistrict n-categories.

7.1.2 Coherence theorem for quasistrict 4-categories

Even though we conjecture that a quasistrict 4-category, as described in Definition 3.2.3, is equivalent to a general weak 4-category, the result has not been proved formally. In comparison with the equivalence result shown for a quasistrict 3-category, a new proof technique would have to be used. In Theorem 3.6.2, we show that a quasistrict 3-category is a **Gray**-category, hence equivalence to a general weak 3-category is obtained due to earlier coherence results for **Gray**-categories. There are no existing coherence results for semistrict and quasistrict 4-categories and no accepted definitions, so a similar shortcut does not exist.

7.1.3 Formalisation of proofs

Methods described in Chapters 2 and 3 are applicable to a wide range of problems in category theory. A straightforward, 'proof of concept' application is to formalise and type check well-established results. In addition to the results already formalised by the community, as discussed in Chapter 4, other examples of results that would be interesting to prove in Globular include for instance:

- Higher-dimensional generalisations of the results on strengthening an adjunction, as shown for $n = 3, 4$ in Theorems 5.0.2 and 5.0.3.
- Results on knotted surfaces embed in $\mathbb{R}^4$ and on knotted 3-manifolds embeded in $\mathbb{R}^5$ [30, 49].
- Algebraic arguments for topological quantum field theories [10].

7.1.4 Sphere eversion

One of the motivating examples for embarking on this research project is formalisation of *sphere eversion*. Informally, a sphere eversion is performed by turning a sphere 'inside-out' in a three dimensional space. It is perhaps counterintuitive at first, how this feat could be achieved using only smooth and continuous transformations, however it is possible and the first proof of existence of such a transformation was provided by Smale [53]. A smooth and continuous transformation is exactly what we expect from a topological point of view, *i.e.*:

- Allowed moves: Bending and contracting surfaces, surfaces crossing each other by 'leaping' into the fourth dimension.
- Disallowed moves: tearing, puncturing, pinching surfaces, creating creases.

The formal definition of the problem is as follows:

Definition 7.1.1 (Sphere eversion)**.** For an embedding of a sphere $f : S^2 \hookrightarrow \mathbb{R}^3$ there exists a regular homotopy of immersions such that $f_0 = f$ and $f_1 = -f$.

The existing constructions of sphere eversions such as those due to Smale [53], Sullivan [57] or Carter [15] consist of hundreds of pictures of 2-dimensional cross-sections of 3-dimensional self-crossing structures. Needless to say, type checking and keeping track of consecutive transformations may get extremely cumbersome in this setting. A model in Globular would allow automatic type-checking and greatly simplify following the individual steps of the proof.

The correct categorical setting in which to model sphere eversion is a symmetric monodial 2-category. We justify intuitively why do we need such a structure which, using the periodic table of higher categories, is a weak 6-category with trivial 0-, 1-, 2- and 3-cells. A monoidal 2-category is necessary to model a 3-dimensional structure. We need braidings to be able to reason about surfaces 'passing through each other', we need syllepsis because we want to regard the two possible orders for such a crossing to be the same. Finally, we need symmetry because we want the two possible ways, in which one crossing can be rewritten into the other, to be the same.

Hence, the necessary condition is to first expand the theory of higher-level singularities in a quasistrict n-category to $n = 5, 6$. An alternative, more crude approach is to model a braided monoidal 2-category as a degenerate quasistrict 4-category with the bottom two levels trivialised, which we can already achieve at this stage, and manual addition of syllepsis and symmetry cells. This is in fact the approach taken by Henriques in the formalisation of sphere eversion that he obtained in Globular. However, a model in a degenerate quasistrict 6-category would offer a higher degree of robustness.

7.1.5 Stochastic processes

Another potential application of Globular is for modelling stochastic processes. In such a process each of the possible events can happen at a time interval with a certain probability. If we let an n-diagram D model the initial state of the process, then every $(n+1)$-cell that rewrites the diagram could be interpreted as an event that the system is subject to. Then, the composite $(n+1)$-cell, *i.e.* a sequence of rewrites, can be treated as a 'history' of events that occurred. If additionally, we assign to each event a rate with which it can happen, we obtain a stochastic process. This feature allows us to abstractly model, for instance, situations arising in nature. There are many biological processes that can be abstractly modelled in this manner, examples include: interactions between populations of wolfs and rabbits, and invasion, infection and spread of viruses throughout a population of cells in a human body.

7.1.6 Properties of higher dimensional rewriting systems

A desirable property of a rewriting system is to be able to decide whether two terms s, t in the system are equal, *i.e.* whether there exists a rewriting path between s and t. In rewriting theory, this is often referred to as the 'word problem' for the rewriting system. There are two properties of a rewriting system that make it simpler to determine existence of rewriting paths between terms. The first is confluence, which means that given a term t any two rewriting paths originating at t meet eventually. This property ascertains that any order the rewriting rules are applied in eventually yields the same result. The second is termination of all rewriting paths, which ensures that there are no infinite sequences of rewrites. In any terminating rewriting system equality is easy to decide, because the normal forms exists for every term t.

As the theory of higher dimensional rewriting is still in early stages of development, these two concepts are not yet well-understood for higher dimensional rewriting paths. It would be interesting to explore them in the context of signatures. Due to the existence of invertible cells, the system is clearly non-terminating. For that reason, the main focus should be the analysis of confluence.

A pair of elements s, r, such that t is rewritable to both, but s and r do not have a common reduct, is called a *critical pair*. Given a non-confluent system in which critical pairs are present, the first step towards creating an equivalent confluent system is to identify all such pairs. Most computational techniques in standard rewriting theory that automatically check for confluence of a rewriting system or perform a Knuth-Bendix completion operate in this manner. Therefore, an analysis of the set of critical pairs in a signature would be the right starting point to get a better understanding of the system. The biggest challenge is that, even though the system might be generated from a finite presentation, the number of critical pairs may be infinite.

Mimram gives a framework in which a finitely generated 3-dimensional rewriting system, based on the notion of an n-polygraph, admits a finite number of critical pairs [42]. It would be of interest to investigate whether a similar framework, which generalises the notion of a critical pair, could be applied to rewriting systems that are finitely presented by the signature structure for $n = 3$ and beyond.

7.1.7 Complementarity

A primary avenue of future work arising from the results in Chapter 6 is investigating the existence of nonstandard models. It has been shown that a category of groupoids, profunctors and spans admits combinatorial 'toy models' of teleportation, as solutions to a 2-categorical equation, from which ordinary quantum teleportation can be recovered by applying a 2-functor into **2Hilb** [8]. It would be interesting to explore whether combinatorial toy models of quantum key distribution can also be built in that setting.

There is also an important open question suggested by Vicary [60]. Recall that in the 2-categorical formalism, we are free to compose the primitive elements in any way that we wish to create abstract specifications of quantum information processing tasks. However, not all those specifications are physically realisable within quantum theory and **2Hilb**. Since specifications remain unchanged by topological manipulation of classical information, realisability is a topological invariant of the specification. The question is, whether this can be deduced by using only topological means.

To conclude, we are certain that studying quantum information processing systems using higher categorical methods will provide a new, abstract point of view at quantum computation. This may lead to being able to show that, in fact, origins of various quantum phenomena lay in the properties of the underlying categorical structure.

Appendices

Appendix A

Logical statements for proofs of correctness of rewriting and composition

Preservation of well-definedness by the process of rewriting a diagram:

Definition A.0.1 ($R(n)$)**.** For $n \geq 0$, let $R(n)$ denote the statement that for any well-defined n-diagrams D, S, T such that S, T are globular with respect to each other, and a well-defined embedding $e : S \hookrightarrow D$, the rewrite $D.\Pi[e, T]$ of D by e is a well-defined diagram.

Definition A.0.2 ($T(n)$)**.** For $n \geq 2$, let $T(n)$ denote the statement that for any well-defined n-diagram D, we have $(D.s).s = (D[i].d).s$ and $(D.s).t = (D[i].d).t$ for any $0 \leq i < |D|$.

Definition A.0.3 ($S(n)$)**.** For $n \geq 1$, let $S(n)$ denote the statement that for any well-defined n-diagrams D, S, T such that $s(S) = s(T)$, $t(S) = t(T)$ and a well-defined n-diagram embedding $e : S \hookrightarrow A$ the following hold:

$$A.\Pi[e, T][j].d =$$
$$= \begin{cases} A[j].d & \text{if } 0 \leq j \leq e.h \\ A[e.h].d.\Pi[e.e, T[j - e.h].d] & \text{if } e.h \leq j \leq e.h + |T| \\ A[j + |S| - |T|].d & \text{if } e.h + |T| \leq j < |A| - |S| + |T| \end{cases}$$

Definition A.0.4 ($Q(n)$)**.** For $n \geq 0$, let $Q(n)$ denote the statement that for any well-defined n-diagrams A, B, C, S, T such that pairs S, T and A, C are globular with respect to each other and for well-defined embeddings $e : S \hookrightarrow A$, $f : C \hookrightarrow B$, the following holds:

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

Definition A.0.5 ($P(n)$)**.** For $n \geq 0$, let $P(n)$ denote the statement that for any well-defined n-diagrams S, T, A, B, C such that pairs S, T and A, C are globular with respect to each other and for well-defined embeddings $e : S \hookrightarrow A$, $f : C \hookrightarrow B$, the following holds for $0 \leq j \leq e.h$:

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

Definition A.0.6 ($B(n)$)**.** For $n \geq 0$, let $B(n)$ denote the statement that for any well-defined n-diagrams S, T, A such that S, T are globular with respect to each other and for a well-defined embedding $e : S \hookrightarrow A$, then the lifted embedding $e.\Lambda[T] : T \hookrightarrow A.\Pi[e, T]$ is well-defined.

Definition A.0.7 ($C(n)$)**.** For $n \geq 0$, let $C(n)$ denote the statement that given two n-diagram embeddings $e : S \hookrightarrow D$ and $f : D \hookrightarrow M$ between well-defined n-diagrams S, D, M, their composite $f \circ e : S \hookrightarrow M$ is well-defined.

Definition A.0.8 ($A(n)$)**.** For $n \geq 0$, let $A(n)$ denote the statement that given three n-diagram embeddings $e : S \hookrightarrow D$, $f : D \hookrightarrow M$, $g : M \hookrightarrow N$ between well-defined n-diagrams S, D, M, N the following equality holds:

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

Preservation of well-definedness by the process of composition of two diagrams:

Definition A.0.9 ($L(k)$)**.** For $k \geq 0$, let $L(k)$ denote the statement that for any well-defined n-diagram D and a well-defined m-diagram S such that $|n - m| = k$ and $t(S) = s^{n-m+1}(D)$ if $m \leq n$ or $t^{m-n+1}(S) = s(D)$ otherwise, then the *composite* diagram $S \circ D$ is well-defined.

Definition A.0.10 ($N(k)$)**.** For $k \geq 0$, let $N(k)$ denote the statement that for any well-defined n-diagram D and a well-defined diagram m-diagram S such that $|n - m| = k$:

- If $n \geq m$ and $t(S) = s^{n-m+1}(D)$ and the composite $S \circ D$ exists, the inclusion embedding $\mathrm{Inc}_r(S, D) : D \hookrightarrow S \circ D$ is well-defined.
- If $n < m$ and $t^{m-n+1}(S) = s(D)$ and the composite $S \circ D$ exists, the inclusion embedding $\mathrm{Inc}_l(S, D) : D \hookrightarrow D \circ S$ is well-defined.

Definition A.0.11 ($K(k)$)**.** For $k \geq 1$, let $K(k)$ denote the statement that for any n-diagram D and any m-diagram S such that $|n-m| = k$ and such that the composite $S \circ D$ exists, the following equalities hold:

$$\begin{aligned} &\text{If } n > m && (S \circ D)[i].d = S \circ (D[i].d) && \text{for any } 0 \leq i < |D| \\ &\text{If } n < m && (S \circ D)[i].d = (S[i].d) \circ D && \text{for any } 0 \leq i < |S| \end{aligned}$$

Definition A.0.12 ($M(k)$)**.** For $k \geq 1$, let $M(k)$ denote the statement that for any well-defined n-diagram D and a well-defined diagram m-diagram S such that $|n - m| = k$ for any $0 \leq i < |D|$ the following equality holds:

$$\begin{aligned} &\text{If } n > m && \mathrm{Inc}_r(S, D[i].d) = (\mathrm{Inc}_r(S, D).e).\Lambda[D[i].d] && \text{for any } 0 \leq i < |D| \\ &\text{If } n < m && \mathrm{Inc}_l(S[i].d, D) = (\mathrm{Inc}_l(S, D).e).\Lambda[S[i].d] && \text{for any } 0 \leq i < |S| \end{aligned}$$

Associativity of diagram composition:

Definition A.0.13 ($E(k)$)**.** For $k \geq 0$, let $E(k)$ denote the statement that for two well-defined n-diagrams D, S, and a well-defined l-diagram M such that $l > n > 0$ and $l - n = k$, the following holds:

$$S \circ (D \circ M) = (S \circ D) \circ M$$

Definition A.0.14 ($F(k)$)**.** For $k, n \geq 0$, let $F(k)$ denote the statement that for two well-defined n-diagrams D, S, and a well-defined l-diagram M such that $l > n$ and $l - n = k$, the following holds:

$$\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M) = \mathrm{Inc}_r(S \circ D, M)$$

Distributivity of diagram composition:

Definition A.0.15 ($G(k)$)**.** For $k \geq 0$, let $G(k)$ denote the statement that for three well-defined diagrams: an n-diagrams D, an m-diagram S and an l-diagram M, such that $l, n > m > 0$ and $|l - n| = k$, the following holds:

$$S \circ_b (D \circ_a M) = (S \circ_b D) \circ_a (S \circ_b M) \qquad \text{if } b < a$$

Here, we have $a = \min(n, l) - 1$, $b = \min(m, \max(n, l)) - 1$.

Definition A.0.16. $[H(k)]$ For $k \geq 0$,let $H(k)$ denote the statement that for three well-defined diagrams: an n-diagrams D, an m-diagram S and an l-diagram M, such that $l, n > m > 0$ and $|l - n| = k$, then, provided that these composites exist, the following holds:

$$\mathrm{Inc}_r(S, D \circ M) \circ \mathrm{Inc}_r(D, M) = \mathrm{Inc}_r(S \circ D, S \circ M) \circ \mathrm{Inc}_r(S, M)$$

Here, we have $a = \min(n, l)$, $b = \min(m, \max(n, l)) - 1$.

Appendix B

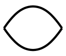
Definition of a switch 3-category

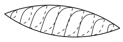
Definition B.0.1 (As given in [22])**.** A switch 3-category T consists of the following two collections of data, subject to axioms as follows:

0-data There are four sets:

· — the set T_0 of objects.

—— — the set T_1 of 1-morphisms.

 — the set T_2 of 2-morphisms.

 — the set T_3 of 3-morphisms.

Moreover there are source and target maps $s, t : T_1 \to T_0$, $s, t : T_2 \to T_1$, and $s, t : T_3 \to T_2$, such that $st = ss$ and $tt = ts$.

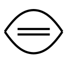
1-data There are thirteen maps of sets in three collections:

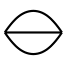
1-morphism target There are two maps with target T_1:

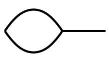
S1-1 : — $i_x : T_0 \to T_1$ — horizontal identity.

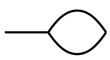
S1-2 : — $m_x : T_1 \times_{T_0} T_1 \to T_1$ — horizontal composition.

2-morphism target There are four maps with target T_2:

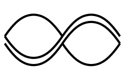
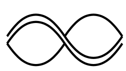
S1-3 :  — $i_y : T_1 \to T_2$ — vertical identity.

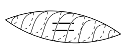
S1-4 :  — $m_y : T_2 \times_{T_1} T_2 \to T_2$ — vertical composition.

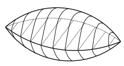
S1-5 :  — $w_r : T_2 \times_{T_0} T_1 \to T_2$ — right whisker.

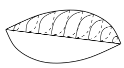
S1-6 :  — $w_l : T_1 \times_{T_0} T_2 \to T_2$ — left whisker.

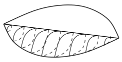
3-morphism target There are seven maps with target T_3. The first of these is a map $sw : T_2 \times_{T_0} T_2 \to T_3$ indicated by depicting the source and target of the 3-morphism $sw(a)$ in terms of the element $a \in T_2 \times_{T_0} T_2$. The remaining six maps we indicate directly by drawing a picture of the resulting 3-morphism.

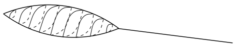
S1-7 :  —  — $sw : T_2 \times_{T_0} T_2 \to T_3$ — switch.

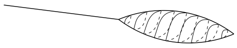
S1-8 :  — $i_z : T_2 \to T_3$ — spatial identity.

S1-9 :  — $m_z : T_3 \times_{T_2} T_3 \to T_3$ — spatial composition.

S1-10 :  — $f_b : T_3 \times_{T_1} T_2 \to T_3$ — bottom fin.

S1-11 :  — $f_t : T_2 \times_{T_1} T_3 \to T_3$ — top fin.

S1-12 :  — $h_r : T_3 \times_{T_0} T_1 \to T_3$ — right 3-cell whisker.

S1-13 :  — $h_l : T_1 \times_{T_0} T_3 \to T_3$ — left 3-cell whisker.

Inverses A 3-morphism $c \in T_3$ is called invertible if there exists a 3-morphism $c^{-1} \in T_3$ such that $m_z(c \times c^{-1}) = i_z(s(c))$ and $m_z(c^{-1} \times c) = i_z(t(c))$. The 1-datum [S1-7] is required to take values in invertible 3-morphisms.

2-axioms The above data are subject to the following thirty-four axioms, together with variant axioms abbreviated in parentheses. In the first fifteen axioms, the condition is that the indicated 1-morphisms or 2-morphisms are equal. In the next four axioms, the conditions is that the 3-morphism obtained by composing all the edges of the diagram is the spatial identity. In the last fifteen axioms, the indicated equation of 3-morphisms is satisfied. There, composition of 3-morphisms denotes spatial composition; also, the variant axioms are indicated as axial reflections of the drawn axioms.

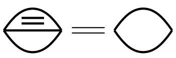
1-morphism axioms

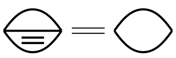
S2-1 :

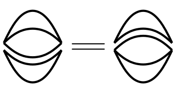
S2-2 :

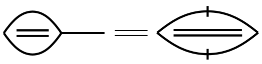
S2-3 :

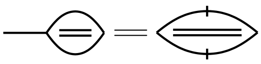
2-morphism axioms

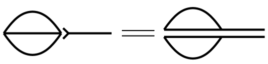
S2-4 : 

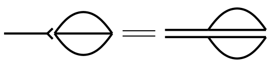
S2-5 : 

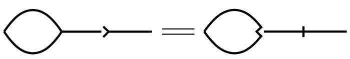
S2-6 : 

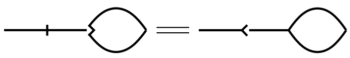
S2-7 : 

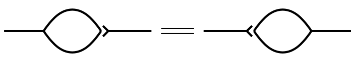
S2-8 : 

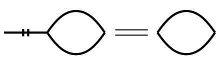
S2-9 : 

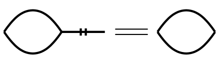
S2-10 : 

S2-11 : 

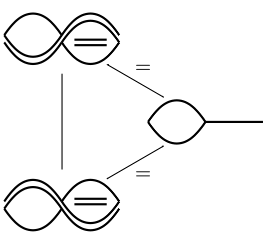
S2-12 : 

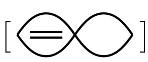
S2-13 : 

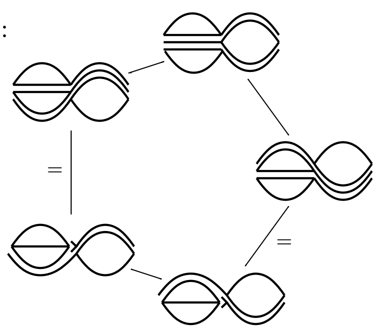
S2-14 : 

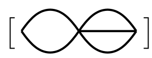
S2-15 : 

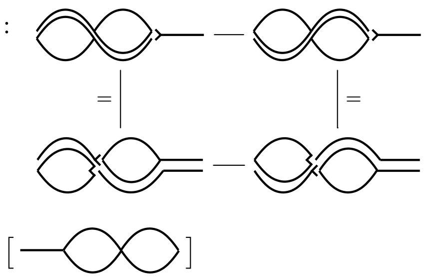
3-morphism axioms

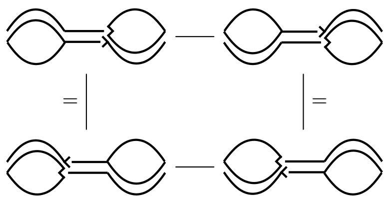
S2-16 : 

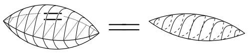


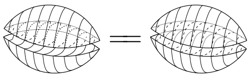
S2-17 : 

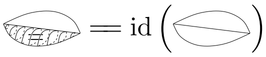


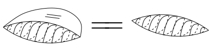
S2-18 : 

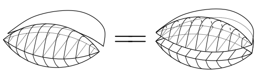
S2-19 : 

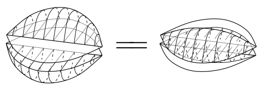
S2-20 : 

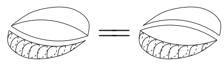
S2-21 : 

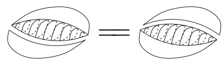
S2-22 : 

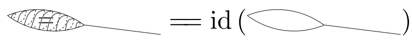
S2-23 : 

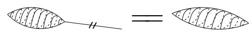
S2-24 : 

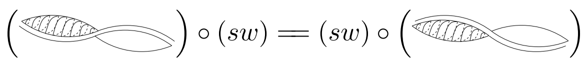
S2-25 : 

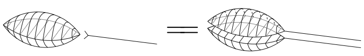
S2-26 : 

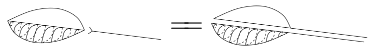
S2-27 : 

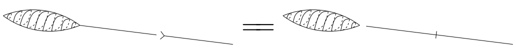
S2-28 : 

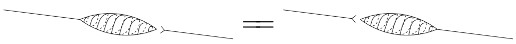
S2-29 : 

S2-30 : 

S2-31 : 

S2-32 : 

S2-33 : 

S2-34 : 

Reflections : z-flip of [S2-20]; y-flip of [S2-22], [S2-23], [S2-24], and [S2-26]; x-flip of [S2-28], [S2-29], [S2-30], [S2-31], and [S2-33]; and x-flip, y-flip, and xy-flip of [S2-32].

In axiom [S2-30], "*sw*" refers to the switch 3-morphism [S1-7].

References

[1] *Handbook of Quantum Logic and Quantum Structures*, volume 2. Elsevier, 2008.

[2] Samson Abramsky and Bob Coecke. A Categorical Semantics of Quantum Protocols. In *Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science, 2004.* Institute of Electrical & Electronics Engineers (IEEE), 2004.

[3] Lev Vaidman, Yakir Aharonov, and David Z. Albert. How to ascertain the values of σ_x, σ_y and σ_z of a spin-$\frac{1}{2}$ particle. 1987.

[4] John Baez and James Dolan. Higher-dimensional algebra and topological quantum field theory. *Journal of Mathematical Physics*, 36(11):6073–6105, 1995.

[5] John Baez and Michael Shulman. Lectures on n-categories and Cohomology, 2006. *arXiv preprint* arXiv:math/0608420.

[6] John C. Baez and Aaron D. Lauda. Higher-dimensional algebra V: 2-groups. *Theory and Applications of Categories*, 12:423–491, 2004.

[7] Krzysztof Bar, Aleks Kissinger, and Jamie Vicary. Globular: a proof assistant for higher-dimensional rewriting. In *1st International Conference on Formal Structures for Computation and Deduction (FSCD 2016)*, volume 52, pages 34:1–34:11, 2016.

[8] Krzysztof Bar and Jamie Vicary. A 2-categorical Analysis of Complementary Families, Quantum Key Distribution and the Mean King Problem. *Electron. Proc. Theor. Comput. Sci.*, 172:316–332, 2014.

[9] John W. Barrett, Catherine Meusburger, and Gregor Schaumann. Gray Categories with Duals and their Diagrams. *arXiv preprint* arXiv:1211.0529, 2012.

[10] Bruce Bartlett, Christopher Douglas, Christopher Schommer-Pries, and Jamie Vicary. Extended 3-dimensional bordism as the theory of modular objects. 2014. *arXiv preprint* arXiv:1411.0945.

[11] C.H. Bennet and G. Brassard. Quantum public key distribution. *IBM Technical Disclosure Bulletin*, 28:3153–3163, 1985.

[12] Laurence Breen. Letter to Michael Hopkins.

[13] Jeffrey Bub. Secure Key Distribution via Pre- and Postselected Quantum States. *Phys. Rev. A*, 63(3), 2001.

[14] Albert Burroni. Higher-dimensional word problems with applications to equational logic. *Theoretical Computer Science*, 115(1):43–62, 1993.

[15] Scott Carter. *An Excursion in Diagrammatic Algebra: Turning a Sphere from Red to Blue*, volume 48 of *Series on Knots and Everything.* World Scientific, rrr edition, 2011.

[16] Eugenia Cheng. Higher-dimensional Category Theory, the Architecture of Mathematics, 2000. Unpublished essay.

[17] Bob Coecke. Quantum Picturalism. *Contemporary Physics*, 51(1):59–83, 2010.

[18] Bob Coecke and Ross Duncan. Interacting Quantum Observables: Categorical Algebra and Diagrammatics. *New J. Phys.*, 13(043016), 2011.

[19] Bob Coecke and Simon Perdrix. Environment and classical channels in categorical quantum mechanics. In *Computer Science Logic*, pages 230–244. Springer Science Business Media, 2010.

[20] Sjoerd Crans. On braidings, syllepses and symmetries. *Doi:* 10.1.1.51.3774, 1998.

[21] Sjoerd Crans. A tensor product for Gray-categories. *Theory and Applications of Categories*, 5(2):12–69, 1999.

[22] Christopher L. Douglas and Andre G. Henriques. Internal Bicategories. *arXiv preprint* https://arxiv.org/abs/1206.4284.

[23] Artur K. Ekert. Quantum cryptography based on Bell's theorem. *Phys. Rev. Lett.*, 67(6):661–663, 1991.

[24] Eric Finster. Opetopic. github.com/ericfinster/opetopic.

[25] Richard Gordon, John Power, and Ross Street. Coherence for Tricategories. *Memoirs of the American Mathematical Society*, 117(558), 1995.

[26] Nick Gurski. *An algebraic theory of tricategories*. PhD thesis, 2007.

[27] Nick Gurski. Biequivalences in Tricategories. *Theory and Applications of Categories*, 26(14):349–384, 2012.

[28] Anne Hillebrand. Quantum Protocols involving Multiparticle Entanglement and their Representations in the zx-calculus. Master's thesis, 2011.

[29] Andre Joyal and Ross Street. The geometry of tensor calculus I. *Advances in Mathematics*, 88:55–112, 1991.

[30] Scott Carter, Seiichi Kamada and Masahico Saito. *Surfaces in 4-Space*. Springer Berlin Heidelberg, 2004.

[31] Aleks Kissinger. *Pictures of Processes*. PhD thesis, University of Oxford, 2011. arxiv.org/abs/1203.0202.

[32] Aleks Kissinger, Alex Merry, Vladimir Zamdzhiev, Lucas Dixon, and Ross Duncan. Quantomatic, 2011. quantomatic.github.io.

[33] Andreas Klappenecker and Martin Rotteler. New Tales of the Mean King. 2005. *arXiv preprint* arxiv.org/abs/quant-ph/0502138v1.

[34] Joachim Kock. *Frobenius Algebras and 2D Topological Quantum Field Theories*. Cambridge University Press (CUP), 2003.

[35] Stephen Lack. A coherent approach to pseudomonads. *Adv. Math.*, 152(2):179–202, 2000.

[36] Saunders Mac Lane. *Categories for the Working Mathematician*. Springer New York, 1971.

[37] Tom Leinster. A survey of definitions of n-category. *Theory and Applications of Categories*, 10(1):1–70, 2002.

[38] Terese Lite. *Term Rewriting Systems*. Cambridge University Press (CUP), 2003.

[39] Shahn Majid. *A Quantum Groups Primer*. Cambridge University Press (CUP), 2002.

[40] MathForum. Perko pair knots. mathforum.org/mathimages/index.php/Perko_pair_knots.

[41] Steffen Metzger. Spin-measurement retrodiction revisited. 2000. *arXiv preprint* arxiv.org/abs/quant-ph/0006115v2.

[42] Samuel Mimram. Towards 3-dimensional rewriting theory. *Logical Methods in Computer Science*, 10(2), 2014.

[43] Bodo Pareigis. *Hopf Algebras in Noncommutative Geometry and Physics*, chapter On Symbolic Computations in Braided Monoidal Categories, pages 269–280. CRC Press, 2004.

[44] Kenneth A. Perko. On the classification of knots. *Proceedings of the AMS*, 45(2):262–262, 1974.

[45] Piotr Pstragowski. On dualizable objects in monoidal bicategories. Master's thesis, Bonn University, 2014.

[46] Daniel G. Quillen. *Homotopical Algebra*. Springer Berlin Heidelberg, 1967.

[47] Emily Riehl and Dominic Verity. Homotopy Coherent Adjunctions and the Formal Theory of Monads, 2013. *arXiv preprint* arXiv:1310.8279.

[48] Saavedra Rivano. *Catégories Tannakiennes*, volume 265 of *Lecture Notes in Mathematics*. Springer Berlin Heidelberg, 1972.

[49] Scott Carter, Masahico Saito. *Knotted Surfaces and Their Diagrams.* American Mathematical Society, 1997.

[50] Peter Selinger. A Survey of Graphical Languages for Monoidal Categories. *New Structures for Physics*, 2009. Lecture Notes in Physics. Springer arXiv:0908.3347.

[51] C. E. Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28(4):656–715, 1949.

[52] Carlos Simpson. Homotopy types of strict 3-groupoids. *arXiv preprint* arXiv:math/9810059, 1998.

[53] Stephen Smale. A classification of immersions of the two-sphere. *Transactions of the American Mathematical Society*, (90):281–290, 1958.

[54] Ross Street. Limits indexed by category-valued 2-functors. *Journal of Pure and Applied Algebra*, (8):149–181, 1976.

[55] Ross Street. Low-dimensional Topology and Higher-order Categories. In *Proceedings of the International Category Theory Conference*, 1995.

[56] Ross Street. *Quantum Groups.* Cambridge University Press (CUP), 2007.

[57] John M. Sullivan. Sphere eversions: From Smale through 'The Optiverse', 1999. *arXiv preprint* arXiv:math/9905020.

[58] Rene Thom. *Structural Stability and Morphogenesis: An Outline of a General Theory of Models.* Addison-Wesley, 1975.

[59] Dominic Verity. Enriched Categories, Internal Categories and Change of Base. *Reprints in Theory and Applications of Categories*, (20):1–266, 2011.

[60] Jamie Vicary. Higher Quantum Theory, 2012. *arXiv preprint* arXiv:1207.4563.

[61] Jamie Vicary and Chris Heunen. Introduction to Categorical Quantum Mechanics, 2012. Set of Lecture Notes.

[62] John von Neumann and M. E. Rose. Mathematical Foundations of Quantum Mechanics. *Phys. Today*, 8(10):21, 1955.

[63] Howard M. Wiseman and Gerard J. Milburn. Quantum Measurement Theory. In *Quantum measurement and control*, pages 1–50. Cambridge University Press (CUP).

[64] Takayuki Yoshida, Masakazu, Miyadera and Hideki Imai. Quantum Key Distribution using Mean King Problem with modified measurement schemes. *Proceedings of 2012 International Symposium on Information Theory and its Applications (ISITA2012)*, 2012.