A unified approach to the construction of categories of games[1]

Nathan James Bowler
Trinity College

August 2010

[1]This dissertation is submitted for the degree of Doctor of Philosophy.

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration.

A unified approach to the construction of categories of games
Nathan Bowler

Our aim is to explain a way to construct categories of games by combining 2-dimensional structures called *playpens*. We use a running example of a category of games based on digraphs.

We explain some basic properties of digraphs, including a construction of a tensor product on this category by means of a promonoidal structure on the category consisting of a parallel pair of maps. We describe a coreflective subcategory of the category of digraphs which is naturally thought of as a category of trees.

To motivate the 2-dimensional structures we shall use, we introduce the concept of unwiring. This formalises the idea of a structure whose elements can be decomposed over the shapes provided by a monad. We show that this construction picks out a well-behaved class of exponentiable objects, and allows the lifting of exponentiation functors to categories of algebras.

We explore the details of the special case of this construction for free multicategory monads. Examining what happens for the free **fc**-mulicategory monad leads to the introduction of playpens. We explore simple constructions and examples of playpens, and show how these may be used to produce an **fc**-multicategory of games based on digraphs.

We outline the theory of representability for **fc**-multicategories, and we demonstrate that the **fc**-multicategory we've produced underlies a strict double category, whose horizontal part is a standard category of games. We explore a link between this construction and slicing in **fc**-multicategories.

We explain how to modify the construction so far to produce categories with different conventions for play, with different notions of strategy, and with different combinatorial notions of game (including trees of the kind mentioned earlier). By means of these examples, we hope to indicate the variety of potential further extensions of our construction.

Contents

Summary **2**

Introduction **5**

Acknowledgements **11**

Primer on multicategories **12**

1 Digraphs and plays **21**

1.1 Digraphs 21

1.2 Promonoidal categories and plain multicategories 25

1.3 Plays and trees 29

2 Unwirings and exponentiability **36**

2.1 Unwirings of objects 37

2.2 Unwirings of maps 42

2.3 Lifting exponentials via unwirings 48

3 Unwirings of multicategories **54**

3.1 Unwirings of maps of multicategories 55

3.2 Playpens 65

3.3 The playpen MAT and the unwirable multicategory RING . . . 74

3.4 Application to digraphs 83

4 Representability **91**
4.1 Opcartesian and weakly opcartesian 2-cells 92
4.2 Composition in playpens 96
4.3 Weakly opcartesian cells in powers of 2 98
4.4 A link to slice constructions 102

5 Basic examples and extensions **109**
5.1 Modifying RING or KNOT 112
5.1.1 Plays of even length 112
5.1.2 Impartial games 114
5.1.3 N-coloured games and Q-coloured games 115
5.1.4 Replacing RING with LINE 117
5.1.5 Slicing 118
5.1.6 Introducing won and lost positions 122
5.2 More familiar notions of strategy 126
5.3 Replacing $\mathbf{Dig}_*$ 133
5.3.1 Tree games 133
5.3.2 Conway games 136

Bibliography **140**

A Playful constructions **142**
A.1 CCM and Pre-colax functors, and playful constructions 143
A.2 Glueing with sections 150

Index **157**

Introduction

The categorical approaches to the understanding of games have so far been almost exclusively 1-dimensional. However, there is a hidden extra dimension of structure underlying the intuitive picture of the relationship between categories and games, and a natural way to make this precise using the fresh language of higher dimensional category theory. In particular, **fc**-multicategories and **fc**-graphs provide a natural framework for constructions both of and in categories of games. This approach raises the possibility of the explicit construction of new games-like categories which had previously only been discussed on an intuitive level. It has also provided a context for the development of a new concept, interesting in its own right; a formal treatment of decomposition over the composition-shapes provided by a cartesian monad.

For a suitably vague understanding of what games are, it is clear that games form the objects of a category with

- maps from the game G to the game H given by second-player strategies for simultaneous play in the dual of G (in which the roles of the two players have been reversed) and in H. The rules for simultaneous play specify that a move should be a move in one or other of the games in question, but not both.

- The identity strategy $1_G \colon G \to G$ given by the copycat strategy 'Always do what your opponent has just done, but in the other game'.

- The composite $g \cdot f$ of $f\colon G \to H$ and $g\colon H \to K$ is more complex. Suppose your opponent makes a move m_0 in G; the following algorithm (which must terminate) gives the appropriate response:
  - f prescribes a move m_1 in response to m_0
  - If m_1 is a move in G then make that move, otherwise g prescribes a move m_2 in response to m_1
  - If m_2 is a move in K then make that move, otherwise f prescribes a move m_3 in response to m_2

  $\vdots$

  A symmetrical procedure provides responses to moves in K.

Formalisations of this idea give mathematics elegant enough to be worth studying for its own sake, but which is also useful in a variety of areas. The categorical construction encodes the basic combinatorics involved in various approaches to the study of games, for example Conway's recursively structured theory in [3], or the specialised theory of hypergraph games, which originated in [6]. These ideas are also applicable to mathematical logic and theoretical computer science. Categories of games have provided some key examples of models of (various fragments of) intuitionistic and linear logic. Such categories have also been used to construct denotational semantics for various abstract programming languages, including the construction of fully abstract models for PCF, as explained in [1] and [8].

The existing constructions have not been fitted into a unified picture. One approach is that of Conway, who gives, in [3], an order relation $\leq$ on games (defined recursively as pairs of sets of games), and proves that this relation is reflexive and transitive. A map from G to H as above could be considered as a constructive proof that $G \leq H$; Conway's (constructive) proofs of reflexivity

and transitivity can be seen as constructing the identities and composition of a category of the kind outlined above. Joyal described this category in [11] (a small modification of Conway's notion of game is necessary to make this work - see section 5.3.2). The use of these ideas in the theory of hypergraph games is less explicit. Hypergraphs are thought of intuitively as having some kind of corresponding game; but usually the intuitive idea of such games (which includes new elements such as the possibility of a draw) is not made formal. Another common approach instantiates games as trees (in this context, a *tree* is a functor $\mathbb{N}^+ \to \mathbf{Set}$). The approach outlined in this document includes all of these cases, as well as a new simple category of games, in which games are instantiated as digraphs. It is this example which we'll use to illustrate the basic construction here, and which will serve as a running example through the rest of the text.

The first key feature of the approach we take in this document is that it involves 2-dimensional structures. This reflects the fact that, in the construction of categories of games, the objects taken as games are usually already combinatorial objects in their own right, with maps-as-combinatorial-objects between them. The vertical dimension of the structures we'll use will model these maps-as-combinatorial objects. It is in the horizontal dimension that the strategies usually taken as maps will emerge. The fact that these horizontal maps form a category will be an emergent property; it will be useful to work with 2-dimensional structures in which the horizontal 1-cells need not be composable.

We'll work principally with 2 kinds of structure satisfying this description. Structures of the first kind, **fc**-multicategories, have been investigated already for very different reasons; they were introduced in [13] as one of a series of kinds of multicategory used in one approach to the definition of weak higher dimensional categories. More recently, they have been used in [4] to provide a general foundation for the theory of multicategories. In that incarnation,

they are referred to as virtual double categories because of their close links to double categories. If an **fc**-multicategory has 2-cells with certain universal properties, then there is a corresponding weak double category (and all weak double categories arise in this way). It is as the horizontal categories of double categories emerging in this way that categories of games will typically arise.

Structures of the second kind, which we have called playpens, are, we believe, new. In place of a 2-dimensional composition operation, they have a kind of decomposition operation which allows 2-cells to be pulled apart over certain kinds of framework. This is a special case of a more general notion of decomposition, which we have christened unwirings. Unwirings have close ties to exponentiability. Recall the definition

DEFINITION 0.0.1. *An object A of a category $\mathcal{E}$ with finite products is* exponentiable *iff the functor $- \times A$ has a right adjoint, $-^A$.*

EXAMPLE 0.0.2. A category with finite products is cartesian closed iff every object is exponentiable.

In particular, any category is exponentiable. **Cat** is not locally cartesian closed, but a complete characterisation of the exponentiable objects in slices of **Cat** is known (see for example [2]). There is a class of objects, the *unwirable* objects (which, intuitively speaking, have both a composition and a decomposition operation which are mutually inverse), which are always exponentiable. Unwirings (which give only the decomposition operation) slightly generalise this - they give a way to lift exponentials to the categories of algebras for certain monads. In particular, exponentiation with respect to playpens lifts in this way. **fc**-multicategories of games typically arise as the exponentials of a simple **fc**-multicategory by a playpen.

The construction of the playpen in question may be modular, in that the combinatorics involved in a typical construction of a category of games can

be separated out into chunks. These chunks are combined via the forming of limits or by gluing along maps of playpens.

As a running example, we'll examine in detail how this can be done for games based on digraphs. A *digraph* consists of a set V of *vertices* and a set E of *edges*, where each edge has a *source* and a *target*, which are vertices assigned to it by maps from E to V. So a digraph is a presheaf on $\bullet \rightrightarrows \bullet$. We'll call the category of digraphs **Dig**. Any digraph may be thought of as the plan of a rudimentary game by considering the vertices as its *positions* and the edges as its *moves*. The category of (pointed) digraphs provides the first chunk of the construction - there is a corresponding playpen whose horizontal structure is essentially trivial.

The second chunk is provided by the functor from digraphs to sets which captures the idea of play in a digraph.

The third chunk is a playpen whose vertical structure is essentially trivial. As it turns out, such playpens are given by unwirable plain multicategories. The unwirable multicategory in question encodes the combinatorics of composition of strategies in the category of games based on digraphs.

The fourth chunk provides a basic combinatorial framework for the kind of combination involved in the idea of 'play in multiple games'. Exponentiation with respect to this playpen is a construction which had already been studied (though the playpen itself had not).

The fifth chunk consists of a map from the third chunk to the fourth. This map encodes the fine detail of the rules of play, and conventions such as impartiality (having the same moves available to each player). A procedure reminiscent of gluing along a functor applied to a combination (by pullback) of the second and fifth chunks gives a playpen incorporating all of this structure. Finally, exponentiating the 2-point lattice by this playpen gives the **fc**-multicategory of games in question.

The document begins with a primer on multicategories, in which we'll

explain the notion of multicategories and the details of plain multicategories and **fc**-multicategories.

In chapter 1, we'll discuss the first and second chunks of this running construction, digraphs and plays, in detail.

In chapter 2, we'll introduce the necessary framework for understanding unwirings, and explain the relationship of unwirings to exponentiability.

In chapter 3, we'll explain how unwirings work in the context of multicategories. We'll introduce the concept of playpens, and show how to build the playpens and maps of playpens which give the remaining 3 chunks of the running construction. I'll also introduce some simple ways to combine playpens (including gluing) and show how these chunks may be fitted together.

In chapter 4 we'll discuss representability, the phenomenon which gives a double category for each **fc**-multicategories with enough universal cells, and show how it applies in the cases we're interested in for the construction of categories of games.

Finally, chapter 5 will contain a range of examples of constructions of and in categories of games, illustrating the wide scope of the techniques introduced in this document.

Acknowledgements

I'm grateful to the EPSRC for providing the money to finance this PhD.

I'd like to thank Stergios Antonakoudis for introducing me to a problem which set off the train of thought leading to the construction in this thesis, Imre Leader for some helpful comments in the very first stages of development of those thoughts, and Richard Garner for a couple of useful discussions at a later stage. I'm extremely grateful to my supervisor, Martin Hyland, who was always eager to listen and made many helpful comments, from minor technical details to broad philosophical points.

Primer on multicategories

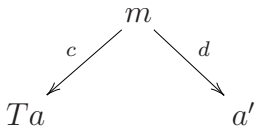
Throughout this document, we shall refer primarily to multicategories in the sense of Leinster. This sense is explained in detail in the first part of the book [13]. In this section, we'll give a brief reminder of the definition of multicategories, and the details of two particular kinds of multicategories, **fm**-multicategories (or *plain* multicategories) and **fc**-multicategories, which will provide the context for many of the constructions of chapter 3. This section is by no means exhaustive, and is no substitute for the fuller treatement of [13].

Intuitively, a multicategory resembles a category. It consists of some objects and some maps, with identities and composition satisfying associativity and identity laws. However, the sources (but not the targets) of maps are allowed to be combinations of objects rather than simply objects. For example, it is natural to consider multilinear maps between vector spaces as maps whose sources are lists of vector spaces. In the general case considered by Leinster, the kind of combination involved is specified by a cartesian monad T. Thus if T is the free monoid monad **fm** on **Set** then the sources of maps are lists of objects; like multilinear maps, the maps can be thought of as taking several input variables.

Definition 0.0.3. *Let T be a cartesian monad on a cartesian category $\mathcal{E}$. The bicategory $\mathcal{E}_{(T)}$ has*

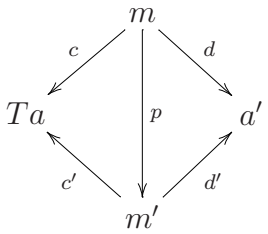
- *objects given by the objects of $\mathcal{E}$.*

- *horizontal 1-cells $a \to a'$ given by spans*



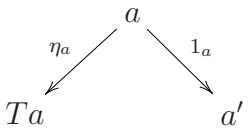
  in $\mathcal{E}$.

- *2-cells $a \overset{m}{\underset{m'}{\Downarrow p}} a'$ given by maps $m \xrightarrow{p} m'$ in $\mathcal{E}$ such that the diagram*



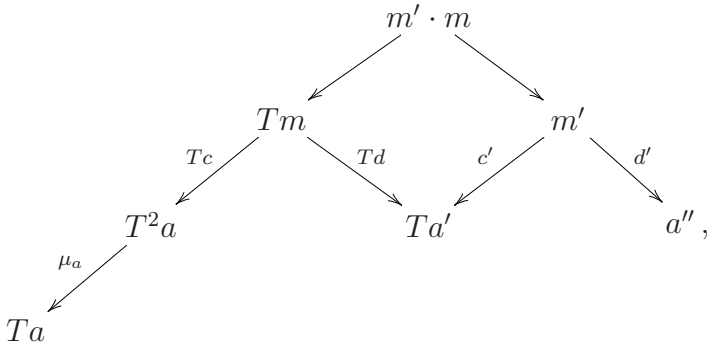
  commutes.

- *the identity 1-cell $a \xrightarrow{1_a} a$ given by the span*



  in $\mathcal{E}$.

- *the composite of $a \xrightarrow{m} a' \xrightarrow{m'} a''$ given by the outer span of the diagram*



  in which the upper square is a pullback.

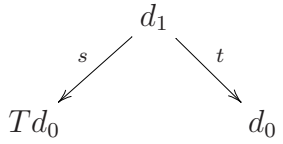
- *the associativity and unit isomorphisms and the horizontal composites of 2-cells induced via the universal property of pullbacks.*
- *the vertical composites of 2-cells given by composition in $\mathcal{E}$.*

A T-multicategory *is a monad in the bicategory* $\mathcal{E}_{(T)}$.

Let's unwind this definition a little, to see how it corresponds to the intuition outlined above. A monad c in $\mathcal{E}_{(T)}$ consists of an object c_0 and a 1-cell $c_0 \xrightarrow{c_1} c_0$ of $\mathcal{E}_{(T)}$, together with 2-cells $c_0 \overset{1_{c_0}}{\underset{c_1}{\Downarrow \mathrm{ids}}} c_0$ and $c_0 \overset{c_1 \cdot c_1}{\underset{c_1}{\Downarrow \mathrm{comp}}} c_0$ satisfying certain equations. Let's focus just on c_0 and c_1 first of all. Together, they form a structure called a T-graph.

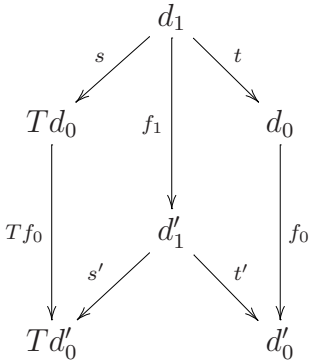
Definition 0.0.4. *The category* T**-Gph** *of* T*-graphs has*

- *objects given by endomorphisms $d_0 \xrightarrow{d_1} d_0$ in $\mathcal{E}_{(T)}$; that is, by spans*



  in $\mathcal{E}$.

- *maps $d \xrightarrow{f} d'$ given by pairs $(d_0 \xrightarrow{f_0} d'_0, d_1 \xrightarrow{f_1} d'_1)$ such that the diagram*

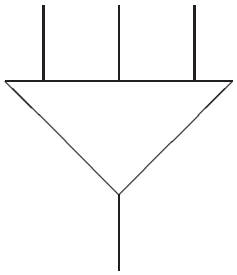


  commutes.

Suppose, for a moment, that $\mathcal{E}$ is **Set**. Then a T-graph d has a set d_0 of objects and a set d_1 of maps. Each map $k \in d_1$ has a target $t(k)$, which is a single object of d, and a source $s(d)$ which is an element of $T(d_0)$ - a combination of objects of d according to the constructions provided by the monad T. The intuitive outline above suggested that a T-multicategory should be a structure of precisely this kind, together with identities and composites of the maps. These identities and composites are provided by the 2-cells ids and comp.

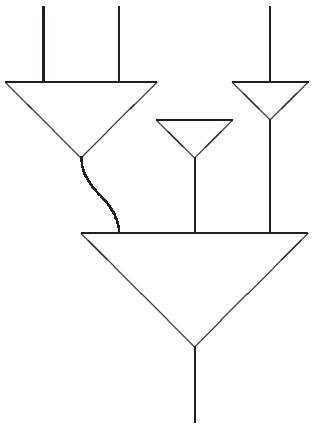
Let's examine how all of this works if T is the identity monad on **Set**. Then a T-graph d is simply a digraph. So a T-multicategory c is a digraph, with vertex set c_0 and edge set c_1, together with a map $c_0 \xrightarrow{\text{ids}} c_1$ picking out an identity edge for each vertex, and a map comp from the pullback of the cospan $c_1 \xrightarrow{t} c_0 \xleftarrow{s} c_1$ - that is, from the set of composable pairs of edges - to c_1. The commutative diagrams these maps must satisfy correpond precisely to the rules about the domains and codomains of identities and composites in a category, and to the associative and identity laws. So, as we might hope, T-multicategories in this case are just categories.

Another example which is worth examining in detail, because it will be important later on, is the case where $\mathcal{E}$ is the category **Set** of sets and T is the free monoid monad **fm**. An **fm**-graph d consists of a set d_0 of vertices together with a set d_1 of edges. Each edge has a *target*, or *output*, which is a vertex, and a *source*, which is an element of $\mathbf{fm}(d_0)$ - that is, a list of vertices. The elements of this list are called the *inputs* of the edge. Such edges are typically drawn with a vertical transistor-like picture, such as



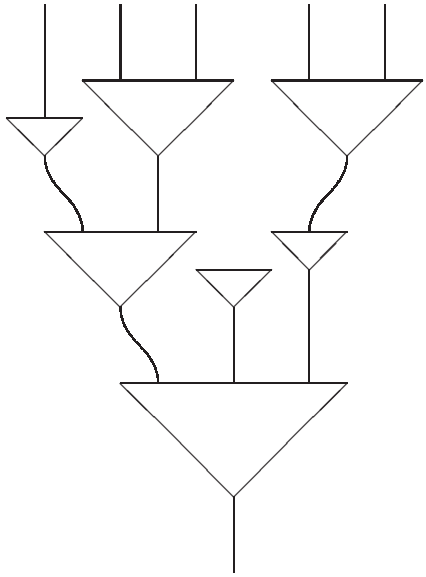
for an edge with three inputs. The number of inputs is called the *arity* of the edge.

An **fm**-multicategory, or *plain* multicategory has in addition an identity map assigning to each vertex (or *object*) a an edge (or *map*) 1_a of arity 1 with a as its unique input and output, and a composition map. The source of the composition map is the set of pairs $((p_i)_{i\in[n]}, p)$, where p is a map of arity n whose inputs are the outputs of the maps p_i. This kind of situation is denoted with a picture like



and called a *composable collection* of maps. Such a composable collection is sent by comp to its *composites* $p \cdot (p_i)_{i\in[n]}$, which is a map with source given by concatenation of the sources of the p_i and target the target of p. These must satisfy identity laws, which say that if each p_i is an identity, then the composite is p, and if p is an identity then the composite is p_1 (in such cases, we must have $n = 1$ as every identity map has arity 1. They must also satisfy

an associative law, which states that the two ways $p \cdot (p_i \cdot (p_{i,j})_{j\in[n_i]})_{i\in[n]}$ and $(p \cdot (p_i)_{i\in[n]}) \cdot (p_{i,j})_{i\in[n],j\in[n_i]}$ of evaluating a composite such as

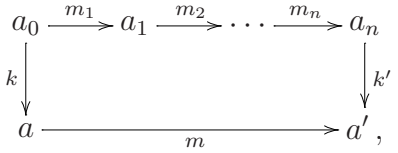


are equal.

EXAMPLE 0.0.5. There is a plain multicategory **Ab** whose objects are abelian groups (or $\mathbb{Z}$-modules) and whose maps with source $(V_i)_{i\in[n]}$ and target V are multilinear maps from $\prod_{i\in[n]} V_i$ to V.

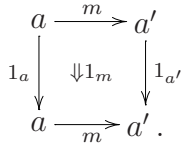
The final example we'll need is a little more complex. It is the case where $\mathcal{E}$ is the category **Dig** of digraphs and T is the free category monad **fc**. An **fc**-graph consists of a pair of digraphs d_0 and d_1. d_0 consists of a set d_{00} of *objects* and a set d_{01} of *horizontal 1-cells*; each such 1-cell has a *source* and a *target*, which are both objects. Horizontal 1-cells are drawn running horizontally from their sources to their targets. d_1 consists of a set d_{10} of *vertical 1-cells* and a set d_{11} of *2-cells*. Each vertical 1-cell also has a *source* and a *target*, which are both objects. Vertical 1-cells are drawn running vertically from their sources to their targets. Each 2-cell has a source and a target in the horizontal direction (which are vertical 1-cells), a source in

the vertical direction (which is a composable collection of horizontal 1-cells) and a target in the vertical direction (which is a horizontal 1-cell). Together, these 1-cells bound a rectangle, such as

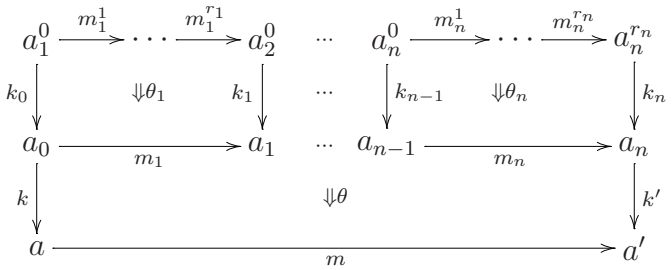


and we say that the 2-cell *fills* this rectangle, and write the name of the cell inside the rectangle to denote this fact.

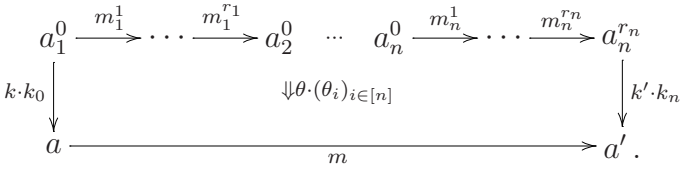
In an **fc**-multicategory, the identity and composite maps restricted to the objects and vertical 1-cells give this digraph the structure of a category. The identity map also gives, for each horizontal 1-cell $a \xrightarrow{m} a'$ a 2-cell



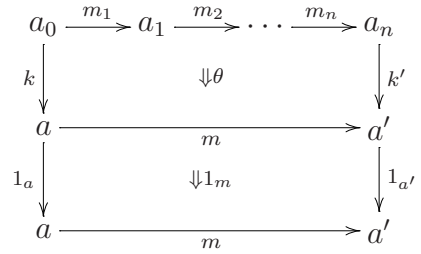
The composition gives, for each *composable collection*



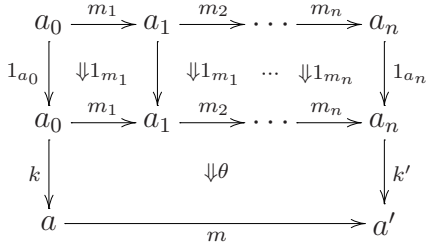
a *composite*



These satisfy identity laws specifying that, for any θ as above, both



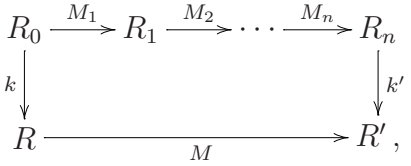
and



compose to give θ.

They also satisfy an associative law, which states that the two possible composites of a diagram like this of depth three are equal. The formula, $\theta \cdot (\theta_i \cdot (\theta_i, j)_{j\in[n_i]})_{i\in[n]} = (\theta \cdot (\theta_i)_{i\in[n]}) \cdot (\theta_i, j)_{i\in[n],j\in[n_i]}$, is the same as that for plain multicategories.

Example 0.0.6. There is an **fc**-multicategory **Mod** whose objects are rings, vertical 1-cells are ring homomorphisms, horizontal 1-cells from R to S are left R-, right S-modules. 2-cells filling the rectangle



are given by multilinear maps θ from $\prod_{i\in[n]} M_i$ to M, which are linear in R_0 and R_n in the sense that

$$k(r).\theta(m_1, m_2, \ldots, m_n).k'(r') = \theta(r.m_1, m_2, \ldots, m_n.r')$$

for $r \in R_0$ and $r' \in R_n$, and linear in each other R_i in the sense that, for $r \in R_i$, $\theta(m_1, \cdots m_i.r, m_{i+1}, \cdots, m_n) = \theta(m_1, \cdots m_i, r.m_{i+1}, \cdots, m_n)$.

For any plain multicategory c there is a corresponding **fc**-multicategory which has just one object and one vertical map - the horizontal 1-cells are given by the objects of c and the 2-cells by the maps of c. Though we shall principally be concerned with **fc**-graphs and **fc**-multicategories, sometimes we will be able to move to discussion of the more familiar plain multicategories via this correspondence.

Example 0.0.7. The full sub-**fc**-multicategory of **Mod** (example 0.0.6) on the object $\mathbb{Z}$ and the vertical map $1_{\mathbb{Z}}$ corresponds to the plain multicategory **Ab** (example 0.0.5).

Chapter 1

Digraphs and plays

To illustrate our construction, we will be using a running example; the construction of a category of games based on digraphs. In this chapter, we'll introduce some of the machinery which will underlie this later construction of a simple category of games. In section 1.1 we'll introduce digraphs and some basic constructions involving them. Then in section 1.2, we'll give a more abstract perspective on the construction of digraphs, and derive some further properties. In section 1.3, we'll explain the crucial notion of a play in a digraph, from which the notion of strategy will later be built. This section will also introduce the relationship between digraphs and trees. Applying our construction to trees instead of digraphs produces essentially the conventional category of games found in, for example, [9].

1.1 Digraphs

A *digraph* D consists of a set D (or $V(D)$ if necessary to avoid ambiguity) of *vertices*, together with a set E (or $E(D)$ if necessary to avoid ambiguity) of *edges*, where each edge has a *source* and a *target*, which are vertices assigned to it by maps s and t from E to D. So a digraph is a presheaf on $\bullet \rightrightarrows \bullet$.

we'll call the category of digraphs **Dig**. E and D extend to functors **Dig** $\to$ **Set**. Since **Dig** is the category of presheaves on $\bullet \rightrightarrows \bullet$, it has all the rich structure of a topos and a locally finitely presentable category (for details, see [10]).

Example 1.1.1. **Dig** has a terminal object 1, with a single vertex $*$ and a single edge @ with source and target at that vertex. This can be represented by the diagram



in which the point represents the vertex and the arrow represents the edge: We will often represent digraphs using diagrams of this kind, or labelled diagrams such as



Example 1.1.2. For any complete-information game (represented however you wish), there is a corresponding digraph with vertices given by the positions in the game, and edges given by the possible moves. So for example Conway's game 0 in [3] corresponds to the digraph with one vertex and no edges. Any digraph may also be thought of as the plan of a rudimentary game: For example the digraph in example 1.1.1 corresponds to the endlessly dull game with just one position, and just one possible move which makes no change in the position.

Accordingly, we'll often refer to the vertices of a digraph as its *positions* and to the edges as its *moves*.

Example 1.1.3. To any set X, there is associated a simple game in which two players take it in turns to claim points from X (once a point has been claimed it cannot be claimed again). This game is the basis for play in the positional game corresponding to any hypergraph structure on X. The associated digraph $\mathrm{Hyp}(X)$ has positions given by pairs (U, V) where U and

V are disjoint subsets of X. U and V are thought of as the sets of points claimed so far by the two players. For each triple (U, V, x), with (U, V) a position and $x \in X$ not in either U or V, there are two corresponding moves. One of them has source (U, V) and target $(U \cup \{x\}, V)$. The other has the same source but has target $(U, V \cup \{x\})$.

Digraphs give a natural formalisation for games of this kind. Other approaches to formalising games confound or separate what we intuitively count as 'the same position'. For example, if we formalise games as trees, then a single position in $\mathrm{Hyp}(X)$ will have multiple representatives in the corresponding tree, one for each history by which it could be reached. If we follow Conway's approach, on the other hand, all positions in which there are k points remaining to be claimed will be identified. This identification would be counterproductive if (as is often the case) we wished to count some of these positions as won for one player and others as won for the other player. This kind of identification also leads to difficulties in the definition of composition of strategies, unless we slightly modify the definition to use families in place of sets, as in section 5.3.2. Another aspect of games played on hypergraphs which conflicts with Conway's approach is that, if the set X is infinite, the digraph $\mathrm{Hyp}(X)$ is illfounded and so cannot be built up recursively.

Since **Dig** is a topos, it already has monoidal structures given by finite products and coproducts. We shall be interested in a third monoidal structure on **Dig**.

Definition 1.1.4. *For digraphs D and D', the vertices of $D \otimes D'$ are given by pairs of vertices, one from D and the other from D'. An edge of $D \otimes D'$ is given by either an edge of D together with a vertex of D' or else by an edge of D' together with a vertex of D. The identity I is given by the digraph $\bullet$ with just one vertex and no edges.*

Example 1.1.5. The tensor product $1 \otimes 1$ of the terminal digraph with itself

is given by the digraph



This is different from both the sum and product of 1 with itself, which are given by



and

respectively.

EXAMPLE 1.1.6. If D and D' are the digraphs corresponding to two games G and G', then $D \otimes D'$ corresponds to the game in which 'G and G' are played simultaneously': It follows the usual conventions for this combination, in which a single move is taken to be a move in either one of the two games being played.

EXAMPLE 1.1.7. If X and Y are sets, $\mathrm{Hyp}(X) \otimes \mathrm{Hyp}(Y) \cong \mathrm{Hyp}(X + Y)$, where $X + Y$ is the disjoint union of X and Y and the function Hyp is that introduced in example 1.1.3.

Implicit in the category **Dig** are notions of subgraph, inverse image, etc. which we'd like to make explicit, and to establish our notation for.

DEFINITION 1.1.8. *Given a set S of positions of a digraph D, the* restriction *$D{\restriction_S}$ of D to S has S as its set of positions, and as moves those of D with both source and target in S; the sources and targets of these moves are preserved in the restriction.*

DEFINITION 1.1.9. *A* subgraph *of a digraph D is a subobject of D in* **Dig**. *That is, it is an isomorphism class of pairs (E, i), with i a monic map of digraphs from E to D. Each such isomorphism class has a canonical representative, in which E is a subset of D and the edges of E form a subset of those of D, with i being given by the injection maps of these subsets (so that the source and target maps of E are given by restrictions of those of*

D). By a standard abuse of notation, we shall also refer to these canonical representatives as the subgraphs of D.

This notion of subgraph is not that usually used by graph theorists, who tend to deal with embedded subgraphs (which correspond to regular monomorphisms in the slightly different categories of graphs they deal with). It is, however, the definition obtained by applying the standard notion of subobject to the category of digraphs, and (more importantly) it is also the notion of subobject which will be useful for discussing the constructions of later chapters.

Definition 1.1.10. *Finally, given a map $f\colon D \to D'$ of digraphs, and a subgraph E of D', the* inverse image $f^{-1}E$ *of E under f is the subgraph of D consisting of all positions or moves which are taken by f to positions or moves in E (this is a pullback in* **Dig** *of the inclusion of E in D' along f).*

1.2 Promonoidal categories and plain multicategories

Tensor products on categories of presheaves are studied in [5], where a method of constructing such tensor products from promonoidal structures is given. In this section, we'll outline how this construction may be used to construct the monoidal category of digraphs. Recall that a *promonoidal* category (or, in the language of [5], a premonoidal category) consists of a category A together with functors $P\colon A^{\mathrm{op}} \times A^{\mathrm{op}} \times A \to \mathbf{Set}$ and $J\colon A \to \mathbf{Set}$ (that is, with profunctors $A \to A \times A$ and $A \to 1$) together with certain natural isomorphisms corresponding to associativity and to left and right identity laws, such that certain diagrams reminiscent of those in the definition of a weak monoidal category commute. This suggests the following

definition (modelled on the definition of an unbiased monoidal category in [13, §3]):

DEFINITION 1.2.1. *An* unbiased *promonoidal category is a pseudo coalgebra for the strict monoidal category pseudocomonad* **FMon** *on the bicategory* **Prof** *of categories, profunctors, and transformations.*

This definition makes sense, since the 2-monad **FMon** lifts in a natural way from **Cat** to give a pseudomonad on **Prof**. Since $\mathbf{Prof} \cong \mathbf{Prof}^{\mathrm{op}}$, there is a corresponding pseudocomonad **FMon** on **Prof** (it makes sense to use the same name, since the underlying functor is the same). This dualisation is a little disconcerting, but it fits the directions of the profunctors used in the original definition: an unbiased promonoidal category consists of a category A together with profunctors $P_n \colon A \to A^n$ together with certain natural isomorphisms, such that 'all diagrams commute'. Copying the proof that the category of unbiased monoidal categories is equivalent to the category of monoidal categories, it is not difficult to show that the category of unbiased promonoidal categories is equivalent to the category of promonoidal categories.

EXAMPLE 1.2.2. Any monoidal category is naturally a promonoidal category; any unbiased monoidal category is naturally an unbiased promonoidal category. The profunctors are given by $P_n(X_i; X)_{i\in[n]} = \mathrm{Hom}_A\left(\bigotimes_{i\in[n]} X_i, X\right)$.

Since for any monoidal category A, $\mathrm{Hom}_A\left(\bigotimes_{i\in[n]} X_i, X\right)$ is the collection of maps of arity n in the underlying plain multicategory[1] of A , this example suggests a further construction:

[1]See [13, §2.1] for the definition of plain multicategories and of the underlying plain multicategory of a monoidal category. We are using a slight variation on the notation Leinster uses there; for a multicategory M, he denotes the collection of maps from the list $(a_1, a_2, ...a_n)$ to the object a by $M(a_1, a_2, ...a_n; a)$, whereas we are denoting the same collection by $M((a_1, a_2, ...a_n), a)$.

Definition 1.2.3. *Let A be be an unbiased promonoidal category, with multiplication profunctors $P_n \colon A \to A^n$. Then the* underlying *plain multicategory $U(A)$ of A has*

- *objects given by the objects of A.*
- *maps with source $(X_i)_{i\in[n]}$ and target X given by elements of the set $P_n((X_i)_{i\in[n]}, X)$.*
- *identities and composition induced from the structural isomorphisms of A.*

Example 1.2.4. If A is a monoidal category, then $U(A)$ is the underlying plain multicategory of A, considered as a monoidal category.

The main result of [5] is that for any promonoidal structure on A, there is a corresponding monoidal structure on $\mathbf{Set}^A$. Essentially the same argument shows that for any unbiased promonoidal structure on A, with multiplication profunctors P_n, there is a corresponding unbiased monoidal structure on $\mathbf{Set}^A$, with the tensor product $\bigotimes_{i\in[n]} F_i$ of the sequence $(F_i)_{i\in[n]}$ of functors given by a coend of the form

$$\int^{(X_i)_{i\in[n]}} \prod_{i\in[n]} F_i(X_i) \times P((X_i)_{i\in[n]}; -)\,.$$

The original promonoidal structure on A may then be recovered by taking $P_n((X_i)_{i\in[n]}, X) = \left(\bigotimes_{i\in[n]} y(X_i)\right)(X)$, where $A^{\mathrm{op}} \xrightarrow{y} \mathbf{Set}^A$ is the Yoneda embedding, and this construction gives a promonoidal structure on A precisely when the tensor product on $\mathbf{Set}^A$ preserves colimits in both variables.

Now we are in a position to explain how this construction gives rise to the monoidal category of digraphs.

EXAMPLE 1.2.5. Recall that **Dig** is isomorphic to $\mathbf{Set}^A$, where A is the category

$$E \underset{t}{\overset{s}{\rightrightarrows}} V \ .$$

Since the tensor product on **Dig** preseves colimits in both variables, it arises as outlined above from some promonoidal structure P on A. For any sequence $(X_i)_{i\in[n]}$ of objects of A, $P_n((X_i)_{i\in[n]}, V) = \left(\bigotimes_{i\in[n]} y(X_i)\right)(v)$, the set of vertices of $\bigotimes_{i\in[n]} y(X_i)$. Now $y(V)$ is the identity digraph I, and $y(E)$ is the digraph $\mathbf{2} = \ s \longrightarrow t$. So the elements of $P_n((X_i)_{i\in[n]}, V)$ are tuples $(\alpha_i)_{i\in[n]}$, with $\alpha_i = 1$ if $X_i = V$ and $(\alpha_i) \in \{s, t\}$ if $X_i = E$. Such a tuple can be thought of as an expression naming a vertex of $\bigotimes_{i\in[n]} D_i$ in terms of vertices and edges of the D_i. For example, if $n = 3$, the tuple $(t, s, 1)$ names the vertex $(t(e_1), s(e_2), v_3)$ in terms of edges e_1 from D_1 and e_2 from D_2 and the vertex v_3 of D_3. If m is the number of values of i with $X_i = E$, these 2^m vertices are the corners of the hypercube framework $\mathbf{2}^{\otimes m}$.

In a similar way, $P_n((X_i)_{i\in[n]}, E)$ is in bijection with the set of $n2^{n-1}$ edges of this hypercube framework. The elements of $P_n((X_i)_{i\in[n]}, E)$ can be thought of as expressions naming edges of $\bigotimes_{i\in[n]} D_i$ in terms of vertices and edges of the D_i. Then the composition of these expressions, considered as cells in the underlying multicategory of A, is given by formal substitution.

The underlying multicategory of A is therefore the multicategory generated by unary cells (s) and (t) from (E) to V, and binary cells $\delta = (1_V, 1_V)\colon (V, V) \to V$, $\partial_L = (1_E, 1_V)\colon (E, V) \to E$ and $\partial_R = (1_V, 1_E)\colon (V, E) \to E$ subject to the relations $s \cdot (\partial_L) = \delta \cdot (s, 1)$, $s \cdot (\partial_R) = \delta \cdot (1, s)$, $t \cdot (\partial_L) = \delta \cdot (t, 1)$, $t \cdot (\partial_R) = \delta \cdot (1, t)$, $\delta \cdot (\delta, 1) = \delta \cdot (1, \delta)$, $\partial_L \cdot (\partial_L, 1) = \partial_L \cdot (1, \delta)$, $\partial_L \cdot (\partial_R, 1) = \partial_R \cdot (1, \partial_L)$ and $\partial_R \cdot (\delta, 1) = \partial_R \cdot (1, \partial_R)$. From this construction in terms of generators and relations, the simple form of the tensor product given in the last section may be derived.

In fact, [5] shows a little more than the existence of a monoidal structure:

the monoidal structures constructed this way are always biclosed, with $F_1 \Rightarrow F_2$ given by

$$\int_{X_2} \mathrm{Hom}_{\mathbf{Set}} \left(\int^{X_1} F_1(X_1) \times P_2((-, X_1), X_2), F_2(X_2) \right) .$$

Example 1.2.6. **Dig** is monoidal closed (since the tensor product in question is symmetric, the left and right closed structures can be identified). The vertices of $D \Rightarrow D'$ are given by maps in **Dig** from D to D'. Given such maps f and g, an edge with source f and target g is given by a map α taking vertices of D to edges of D' such that for any vertex x, $s(\alpha(x)) = f(x)$ and $t(\alpha(x)) = g(x)$. These edges are reminiscent of natural transformations between the maps, but of course it makes no sense in this context to impose a condition that 'naturality squares commute'.

1.3 Plays and trees

In thinking of a digraph D as a game, it is helpful to consider the set $\mathbf{Pl}(D)$ of possible plays in that game, a *play* being a sequence of moves which could be played in succession. The quickest way to make this precise is to define a play in D to be a map in the free category $\mathbf{fc}(D)$ on D. More explicitly, a *play* p of length $n > 0$ with source s and target t in a digraph D is a sequence $(p_i)_{i \in [n]}$ of moves from D with the property that $s(p_1) = s$, $s(p_{i+1}) = t(p_i)$ for $i \in [n-1]$ and $t(p_n) = t$. A play p of length 0 with source s and target t only exists if $s = t$, and then there is only one, denoted by 1_s.

Example 1.3.1. A play in the terminal digraph 1 (example 1.1.1) consists of a sequence of @s of length some natural number[2] $n \in \mathbb{N}_0$. This gives an

[2]Traditionally, the set $\mathbb{N}$ of natural numbers can be taken to either include or exclude 0. Since we want to make a distinction between these options, we shall refer to the set of natural numbers including 0 as $\mathbb{N}_0$ and the set of natural numbers excluding 0 as $\mathbb{N}^+$.

identification between $\mathbf{Pl}(1)$ and $\mathbb{N}_0$. This is not surprising, since a 1-object category is a monoid and the free monoid on the terminal set is $\mathbb{N}_0$.

Any map of digraph games $f\colon G \to H$ induces a functor on the free categories, and so induces a function on plays (by acting termwise). We will denote this function by f_*.

Example 1.3.2. For any digraph D, the unique map $!\colon D \to 1$ induces the map $!_*\colon \mathbf{Pl}(D) \to \mathbb{N}_0$ taking each play to its length.

Set, being a category, can be considered as a digraph.

Definition 1.3.3. *Given a digraph D, an* algebra *for D is a map of digraphs $D \to \mathbf{Set}$.*

There are many alternative formulations of the idea of an algebra for a digraph. First, explicitly, an algebra A is given by a set $A(x)$ for each vertex x of D, and a function $A(e)\colon A(s(e)) \to A(t(e))$ for each edge e of D. Second, since **Set** is a category, algebras for D (as a digraph) correspond to algebras for $\mathbf{fc}(D)$ (as a category), that is, to functors $\mathbf{fc}(D) \to \mathbf{Set}$.

Example 1.3.4. An algebra for the terminal digraph 1 is given by a set X together with an endomorphism f of that set. This corresponds to an action of the monoid $\mathbb{N}_0$ on X: The action of n is given by f^n.

The variety of formulations of the concept of an algebra for a category now suggest a couple of other reformulations of the idea of an algebra for a digraph. For example, to each algebra A for a digraph D there is a corresponding discrete opfibration $\mathbf{Ar}(A) \to \mathbf{fc}(D)$, where $\mathbf{Ar}(A)$ is the category of arrows of A.[3] There is also a corresponding discrete opfibration $\mathbf{Ar}_d(A)$ over D in **Dig**, given by pulling back along the map $D \to \mathbf{fc}(D)$ in **Dig**.

[3]The correspondence between algebras for a category and discrete opfibrations over it is categorical folklore. For a typical treatment see [13, §1].

The original map A may be recovered from $\mathbf{Ar}_d(A)$; each vertex x of D gets sent to the set $A(x)$ of its preimages in $\mathbf{Ar}_d(A)$, and $A(e)(y)$ is given by the unique object $y' \in A(t(e))$ such that there is an edge from y to y' in $\mathbf{Ar}_d(A)$.

EXAMPLE 1.3.5. A discrete opfibration over 1 is given by a digraph such that for each vertex there is a unique edge with source at that vertex. Thus the collection of edges forms the graph of a function f from the set X of vertices to itself.

Another reformulation: There is a typed algebraic theory $\mathcal{T}_D$ with one type for each vertex of D and one unary operation for each edge of D: Then algebras for D correspond to algebras for $\mathcal{T}_D$.

EXAMPLE 1.3.6. The theory $\mathcal{T}_1$ has a single type $*$ and a single unary operation @. An algebra for this theory is therefore given by a set together with an endomorphism on that set.

So far, we have considered plays that can start and end anywhere in a digraph. However, in the normal intuitive conception of games, it is traditional to have a chosen starting position from which all plays should begin. To capture this notion, we shall have to work with pointed digraphs, with the basepoint thought of as giving the initial position from which the game is to be played.

EXAMPLE 1.3.7. For any set X, it is usual to think of play in the digraph $\mathrm{Hyp}(X)$ of example 1.1.3 as beginning at $(\emptyset, \emptyset)$, so we shall take this as a basepoint for $\mathrm{Hyp}(X)$.

An algebra for a pointed digraph (D, x) is given by a pair (A, a), with A an algebra for D and $a \in D(x)$, or equivalently by a discrete opfibration over (D, x) in $\mathbf{Dig}_*$ or an algebra for the theory $\mathcal{T}_{D,x}$ obtained from $\mathcal{T}_D$ by adding a single constant a of type x. The free algebra for (D, x) will correspond

(by the Yoneda lemma) to the functor $A_{D,x} = \mathbf{fc}(D)(x, -)\colon \mathbf{fc}(D) \to \mathbf{Set}$. Explicitly, $A_{D,x}(y)$ is the set of all plays with source x and target y, and $A_{D,x}(e)$ is the function 'adjoin e as the final term'. The value of the constant a in $A_{D,x}$ is given by 1_x in $\mathbf{fc}(D)$, that is, by the empty sequence of moves from x to x.

EXAMPLE 1.3.8. The free algebra for $\mathcal{T}_{(1,*)}$ has a single object $f^n(a)$ for each $n \in \mathbb{N}_0$. This algebra corresponds to the Yoneda functor $\mathbb{N}_0 \to \mathbf{Set}$ sending the unique object to $\mathbb{N}_0$ and each natural number n to the operation 'add n'. In particular, the free algebra can be identified with $\mathbb{N}_0$, with the single constant a given by 0 and f acting as 'add 1'.

The corresponding opfibration of (D, x) in $\mathbf{Dig}_*$ consists of the pointed digraph $U(D, x)$ with vertices given by plays in D with source x, with a unique edge from p to p' if p' extends p by a single move, and with basepoint 1_x. The map $\epsilon_{D,x}$ from this pointed digraph to (D, x) takes each play to its target and each edge from p to p' to the edge of D which was adjoined to p to give p'.

EXAMPLE 1.3.9. The opfibration $U(1)$ of 1 corresponding to A_* consists of the digraph

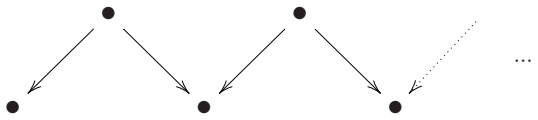
with vertex set $\mathbb{N}_0$ and a unique map from n to $n + 1$ for each n.

The map U constructed above extends in an obvious way to a functor $U\colon \mathbf{Dig}_* \to \mathbf{Dig}_*$, making ϵ a natural transformation $U \to 1$. Intuitively, $U(D)$ can be thought of as 'D played with a notebook keeping track of the history' (here positions with different records in the notebook are counted as different). Then $U^2(D)$ would correspond to 'D played with 2 notebooks, the first keeping track of history and the second keeping track of what has been written in the first': The second notebook here appears to be adding

no information, so one might expect $\epsilon_{U(D),x} \colon U^2(D) \to U(D)$ to be an isomorphism. That it is an isomorphism can be seen by observing that $U(D)$ is initial in the slice of the category of pointed digraphs and discrete opfibrations (of pointed digraphs) by D, and that initiality is preserved by taking the slice.

It follows from the comments in the last paragraph that U, together with ϵ and the inverse of $U\epsilon$, forms an idempotent comonad on $\mathbf{Dig}_*$. In this situation, the category of coalgebras is equivalent to the full subcategory on the objects of the form $U(D)$. Despite this 'normal form' theorem, it is still worth looking at what the coalgebras of U in this case actually are.

Consider some coalgebra $T \xrightarrow{h} UT$. Composing h with the map $U!$ gives a map $T \to U1$, by means of which the set of positions of T may be decomposed as a disjoint union $\bigsqcup_{n \in \mathbb{N}_0} T_n$. Any move in T must then go from an element of T_n to an element of T_{n+1}, for some n. Thus T may be considered as a diagram of shape

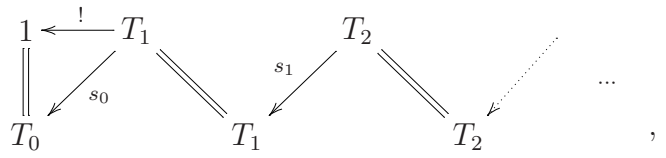


in **Set**. UT may also be considered as a diagram of this shape, in which the left hand object is terminal and all the right-pointing arrows are isomorphisms. But h and ϵ_T pick out T as a retraction of UT, so that T also has these properties. That is, T may be considered as a diagram of shape

in **Set**. Conversely, any such diagram

$$T_1 \xleftarrow{s_1} T_2 \xleftarrow{s_2} T_3 \xleftarrow{s_3} T_4 \longleftarrow \cdots$$

may be extended to



giving a digraph $d(T)$ for which the positions are given by $\bigsqcup_{n\in\mathbb{N}_0} T_n$, the moves are given by $\bigsqcup_{n\in\mathbb{N}^+} T_n$, the source map is given by $\bigsqcup_{n\in\mathbb{N}_0} s_n$ and the target map is given by the identity. This digraph can be given the structure of a U-coalgebra in a unique way.

Now, diagrams of shape

in **Set** are a standard implementation of trees , and so we shall refer to the category $\mathbf{Set}^{(\mathbb{N}^+)^{\mathrm{op}}}$ of such diagrams as **Tree**. What the last few paragraphs show is that **Tree** is equivalent to the category of coalgebras for U, with adjoint functors $d\colon \mathbf{Tree} \to \mathbf{Dig}_*$, as described in the last paragraph, and $t\colon \mathbf{Dig}_* \to \mathbf{Tree}$ with $t(D)_n$ given by the set of all plays from the basepoint in D of length n, and all the maps given by truncation. Since the comonadic adunction $d \dashv t$ is a coreflection and interacts well with the symmetric monoidal structure on $\mathbf{Dig}_*$ ($t(\epsilon \otimes \epsilon)\colon t(dtA \otimes dtB) \to t(A \otimes B)$ is an isomorphism for all digraphs A and B), there is as in [12] a unique symmetric monoidal structure on **Tree** making (the opposite of) $d \dashv t$ a symmetric monoidal adjunction. Specifically, the identity is given by $t(I)$ and the tensor product of A and B is given by $t(d(A) \otimes d(B))$ and with respect to these structures t is strong monoidal and d is colax monoidal.

Trees of this kind are often considered as an implementation of games: Intuitively, for a tree T, the points of the sets T_n are thought of as being positions of the game. It is possible for the n^{th} move to go from $p \in A_n$ to $p' \in A_{n+1}$ precisely when $f_n(p') = p$. Reassuringly, this intuition is captured by the functor d. We shall return to this point of view in chapter 5.

An alternative point of view on the functor U is that it picks out the possible plays in a game, beginning at the basepoint. Specifically, composing V, the functor sending any pointed digraph to its set of vertices, with U gives the colax monoidal functor

$$\mathbf{Dig}_* \xrightarrow[U]{} \mathbf{Dig}_* \xrightarrow[V]{} \mathbf{Set} \qquad \mathbf{Play}: \mathbf{Dig}_* \to \mathbf{Set}$$

sending any pointed digraph to the set of plays from the basepoint in that digraph. This functor will provide some key data for our running example of a construction of a category of games.

Chapter 2

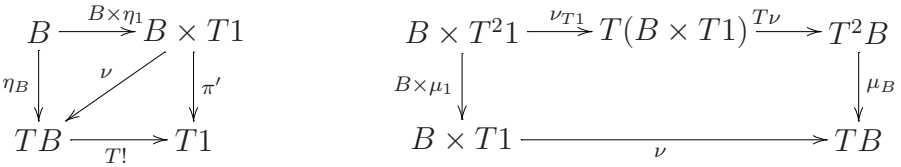
Unwirings and exponentiability

The key concept for this chapter is that of *unwirings* with respect to a cartesian monad T. If the monad T is thought of as providing some constructions by which things may be combined, an unwiring of an object gives ways to take the things in it apart over the same constructions. An object with an unwiring can therefore be thought of as a reverse algebra, or *arbegla*. This reversal often appears in the examples; for example, where algebras correspond to lax maps of a particular kind, arbeglas correspond to colax maps. However, arbeglas correspond to a intuition for deconstruction quite unlike that for coalgebras.

In section 2.1, we'll introduce unwirings and arbeglas, and give some of their basic properties and some very simple examples. In section 2.2, we'll extend the notion of unwiring from objects to maps, and explore the new perspective this brings and the new examples which arise. Finally, we'll explain the relationship between unwirings and exponentials in the section 2.3.

2.1 Unwirings of objects

Definition 2.1.1. *Let T be a cartesian monad on a category $\mathcal{E}$ with finite limits, and B an object of $\mathcal{E}$. An* unwiring *of B is a map $\nu\colon B \times T1 \to TB$ making the follwing diagrams commute:*



where ν_{T1} is determined by its components $(\nu \cdot (B \times T!), \pi')$ with respect to the pullback $T(B \times T1) = TB \times_{T1} T^2 1$. If it is necessary to specify the monad in question, we'll call ν an unwiring with respect to T.

Intuitively, the monad T may be thought of as specifying constructions by means of which things can be combined together; $T1$ consists of all the shapes that such constructions might take. An unwiring of B, on the other hand, should be thought of as a way to decompose the things in B according to the same constructions. For anything in B, and any shape of construction which might potentially have produced that thing, the unwiring specifies a way to take that thing apart into a collection of bits (called its *decomposition*) which might fit together in that shape of construction. The commutative diagrams that must be satisfied specify that this way of taking things apart must behave in a sensible manner. This intuitive account should not be taken as rigorous; in some contexts, it may not even be sensible to talk about elements of B.

Example 2.1.2. If T is the identity monad on $\mathcal{E}$, then the only unwiring of B is the canonical isomorphism $B \times 1 \to B$.

Example 2.1.3. Denote the free monoid monad on **Set fm**. Then **fm**1 is $\mathbb{N}_0$ and so any unwiring ν of a set B can be decomposed into components

$\nu_n \colon B \to TB$. The commutativity of the lower triangle in the first diagram says that the image of ν_n consists of sequences of length n. The commutativity of the upper triangle says that $\nu_1(b) = (b)$ for each $b \in B$, and the commutativity of the rectangle on the right says that for any sequence $(n_i)_{i\in[n]}$ and any $b \in B$, $\nu_{\sum_{i\in[n]} n_i}(b)$ is obtained by concatenating the sequences $\nu_{n_i}(\nu_n(b)_i)$. Now apply this to the sequence δ^n_j of length n in which all terms except the j^{th} are 0 and the j^{th} term is 1. For any b, n and i, $(b) = \nu_1(b)$ is the sequence obtained by concatenating the $\nu_{\delta^n_{ji}}(\nu_n(b)_i)$, which is the sequence $(\nu_n(b)_i)$.

That is, the only unwiring ν of any set B is the one with $\nu_n(b)$ the constant sequence of length n with all terms equal to b.

Example 2.1.4. Let T and $\mathcal{E}$ be as in definition 2.1.1, and let $\mathcal{C}$ be any category. The monad T lifts to a monad $T^{\mathcal{C}}$ on the category $\mathcal{E}^{\mathcal{C}}$. An unwiring of a functor $B \colon \mathcal{C} \to \mathcal{E}$ with respect to $T^{\mathcal{C}}$ is given by a natural transformation $\nu \colon B \times T1 \to TB$ satisfying certain conditions. These conditions reduce to the statement that, for each object C of $\mathcal{C}$, ν_C is an unwiring of $B(C)$. Thus an unwiring of a functor B consists of unwirings ν_C of $B(C)$ for each C, making certain naturality squares commute.

This suggests the following definition, which plays on the fact that unwirings give a reversal of the notion of T-algebras.

Definition 2.1.5. *Let T and $\mathcal{E}$ be as in definition 2.1.1. The category T-***Arb** *of T-arbeglas has*

- *objects given by pairs (B, ν), where ν is an unwiring of B.*
- *maps $(B, \nu) \to (B', \nu')$ given by maps $B \xrightarrow{f} B'$ in $\mathcal{E}$ such that ν and ν' give an unwiring of f with respect to $T^{\mathbf{2}}$. This condition is equivalent*

to commutativity of the diagram

$$\begin{array}{ccc} B \times T1 & \xrightarrow{\nu} & TB \\ {\scriptstyle f\times T1}\downarrow & & \downarrow{\scriptstyle Tf} \\ B' \times T1 & \xrightarrow[\nu']{} & TB' \,. \end{array}$$

Consideration of example 2.1.4 now gives the following proposition:

PROPOSITION 2.1.6. *Let T, $\mathcal{E}$ and $\mathcal{C}$ be as in example 2.1.4. In all such cases, $(T\textbf{-Arb})^{\mathcal{C}} \cong T^{\mathcal{C}}\textbf{-Arb}$.* □

The map $T \mapsto T\textbf{-Arb}$ can now be extended to a functor **Arb** from the category of monads on categories with finite limits and weak limit preserving maps of monads to the category of categories.

PROPOSITION 2.1.7. *Let T and T' be cartesian monads on the categories $\mathcal{E}$ and $\mathcal{E}'$, and suppose that $\mathcal{E}$ and $\mathcal{E}'$ have finite limits. Let (Q, ψ) be a weak (resp. strict) map of monads $T \to T'$ in which Q preserves finite limits. Let (f, ν) be an unwiring of B with respect to T. Then the map $\nu' = \psi_B^{-1} \cdot Q\nu \cdot (QB \times \psi_1)$ (resp. $Q\nu$) is an unwiring of QB with respect to T'.*

Proof. We shall only prove the third identity; the remaining proofs are similar but simpler. Consider the diagrams

$$\begin{array}{ccccc} QB \times T'^2Q1 & \xrightarrow{QB\times\mu_{Q1}} & QB \times T'Q1 & \xrightarrow{\nu'} & T'QB \\ {\scriptstyle QB\times(\psi_{T1}\cdot T'\psi_1)}\downarrow & & {\scriptstyle QB\times\psi_1}\downarrow & & \downarrow{\scriptstyle \psi_B} \\ QB \times QT^2 1 \times Q\mu_1 & \xrightarrow[QB]{} & QB \times QT1 & \xrightarrow[Q\nu]{} & QTB \end{array}$$

and

$$\begin{array}{ccccccc}
QB \times T'^2Q1 & \xrightarrow{\nu'_{TQ1}} & T'(QB \times T'Q1) & \xrightarrow{Q\nu'} & T'^2QB & \xrightarrow{\mu_{QB}} & T'QB \\
\downarrow{\scriptstyle QB\times T'\psi_1} & & \downarrow{\scriptstyle T'(QB\times\psi_1)} & & \downarrow{\scriptstyle T'\psi_B} & & \\
QB \times T'QT1 & \xrightarrow{\xi} & T'Q(B \times T1) & \xrightarrow[T'Q\nu]{} & T'QTB & & \downarrow{\scriptstyle \psi_B} \\
\downarrow{\scriptstyle QB\times\psi_{T1}} & & \downarrow{\scriptstyle \psi_{B\times T1}} & & \downarrow{\scriptstyle \psi_{TB}} & & \\
QB \times QT^2 1 & \xrightarrow[Q\nu_{T1}]{} & QT(B \times T1) & \xrightarrow[QT\nu]{} & QT^2B & \xrightarrow[Q\mu_B]{} & QTB\,,
\end{array}$$

in which the map ξ is determined by its components $\langle \nu' \cdot (QB \times T'!), \pi' \rangle$ with respect to the pullback $T'QB \times_{T'Q1} T'QT1 = T'(QB \times QT1) = T'Q(B \times T1)$. Both diagrams commute, and the bottom and side composites of the rectangles are equal. Thus since the left side of each rectangle is an isomorphism, the top composites are also equal, as required. □

Definition 2.1.8. *Let $\mathcal{E}$, T, $\mathcal{E}'$, T', Q and ψ be as in proposition 2.1.7. Then the functor $Q_* = (Q, \psi)$-**Arb**: T-**Arb** $\to$ T'-**Arb** acts by*

- *sending the T-arbegla (B, ν) to $(QB, \psi_B^{-1} \cdot Q\nu \cdot (QB \times \psi_1))$*
- *sending the map f of T-arbeglas to Qf.*

Example 2.1.9. V is a strict map of monads from the free category monad **fc** on **Dig** to the identity monad on **Set**. So if (B, ν) is an **fc**-arbegla, then $(V(B), V(\nu))$ is a 1-arbegla. Thus as in example 2.1.2 $V(\nu)$ must be the identity map. As in example 2.1.3, ν can be split up into maps $\nu_n \colon B \to \mathbf{fc}(B)$, with ν_n sending each edge of B to a composable string of edges of length n, with the same source and target. In particular, ν_0 sends each edge of B to an identity map, in which the source and target are identical. It follows that, for each edge in B, the source and target of that edge are identical. So the edges may be divided up into sets B_b, indexed by the vertices of B, such that the source and target of every edge in B_b is b. Then ν_n sends elements of B_b to elements of $\mathbf{fm}(B_b)$, where **fm** is the free monoid monad.

That is, ν may alternatively be split into maps $\nu_b \colon B_b \times \mathbf{fm}(1) \to \mathbf{fm}(B_b)$. The conditions that ν is an unwiring with respect to **fc** now precisely state that each ν_B is an unwiring with respect to **fm**. In summary, **fc**-arbeglas correspond to indexed families of **fm**-arbeglas. But it is clear from example 2.1.3 that **fm**-arbeglas correspond to sets. Therefore **fc**-arbeglas correspond to indexed families of sets.

PROPOSITION 2.1.10. *If an unwiring ν of B is an isomorphism, then $(B, \pi \cdot \nu^{-1})$ is a T-algebra. This gives a bijection between unwirings which are isomorphisms and T-algebras (B, β) with the property that $B \xleftarrow{\beta} TB \xrightarrow{T!} T1$ is a product diagram.*

Proof. First, we should check that $\pi \cdot \nu^{-1}$ satisfies the algebra identities. $\pi \cdot \nu^{-1} \cdot \eta_B = \pi \cdot (B \times \eta_1) = 1_B$ and $\pi \cdot \nu^{-1} \cdot \mu_B = \pi \cdot \nu^{-1} \cdot \mu \cdot T\nu \cdot \nu_{T1} \cdot \nu_{T1}^{-1} \cdot T\nu^{-1} = \pi \cdot (B \times \mu) \cdot \nu_{T1}^{-1} \cdot T\nu^{-1} = \pi \cdot \nu_{T1}^{-1} \cdot T\nu^{-1} = \pi \cdot (B \times T!) \cdot \nu_{T1}^{-1} \cdot T\nu^{-1} = \pi \cdot \nu^{-1} \cdot T\pi \cdot T\nu^{-1}$. Further, ν is the canonical isomorphism identifying $B \xleftarrow{\pi \cdot \nu^{-1}} TB \xrightarrow{T!} T1$ as a product diagram.

It remains to check that any T-algebra (B, β) with this property arises in this way. Without loss of generality, $B \xleftarrow{\beta} TB \xrightarrow{T!} T1$ is the canonical representation of $B \times T1$, so that ν is 1_{TB}. Then $\beta \cdot \eta_B = 1_B$ and $T! \cdot \eta_B = \eta_1 \cdot !$ and so $\eta_B = B \times \eta_1$, so the left diagram commutes. But also $\beta \cdot \mu_B \cdot T\nu \cdot \nu_{T1} = \beta \cdot T\beta \cdot T\nu \cdot \nu_{T1} = \beta \cdot T\pi \cdot \nu_{T1} = \beta \cdot \nu \cdot (B \times T!) = \pi \cdot (B \times T!) = \pi$ and $T! \cdot \mu_B \cdot T\nu \cdot \nu_{T1} = \mu_1 \cdot T^2! \cdot T\nu \cdot \nu_{T1} = \mu_1 \cdot T\pi' \cdot \nu_{T1} = \mu_1 \cdot \pi'$, and so the right hand diagram also commutes. □

Accordingly, we'll call T-algebras B such that $B \xleftarrow{\beta} TB \xrightarrow{T!} T1$ is a product diagram *unwirable* T-algebras.

EXAMPLE 2.1.11. Only the terminal monoid is unwirable. A category is unwirable iff it is discrete.

This poverty of examples, which holds for most of the usual monads we consider on **Set**, is a little discouraging. One reason why there are so few

examples over **Set** will be explained in remark 2.3.2. Nevertheless, there are many interesting examples, which we'll be considering later (especially in chapter 3). These are typically arbeglas for certain 'free multicategory' monads. For example, the unwirable plain multicategories will turn out to correspond to discrete promonoidal categories.

2.2 Unwirings of maps

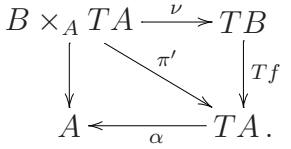
A natural generalisation of the notion of an unwiring of an object is the notion of an unwiring of a map.

Let T be a cartesian monad on a cartesian category $\mathcal{E}$. So for this section $\mathcal{E}$ is not required to have a terminal object unless this is specified, but only pullbacks. Let (A, α) be a T-algebra. Then the monad T lifts to a cartesian monad T/A on $\mathcal{E}/A$ which sends $B \xrightarrow{f} A$ to $TB \xrightarrow{Tf} TA \xrightarrow{\alpha} A$. Since $\mathcal{E}/A$ has all finite limits, it is possible to consider unwirings in this context. An unwiring of (B, f) is a map $\nu\colon B \times_A TA \to TB$ making the following diagrams commute in $\mathcal{E}$:

$$\begin{array}{ccc} B & \xrightarrow{B\times_A \eta_A} & B \times_A TA \\ {\scriptstyle \eta_B}\downarrow & \swarrow_{\nu} & \downarrow{\scriptstyle \pi'} \\ TB & \xrightarrow[Tf]{} & TA \end{array} \qquad \begin{array}{ccccc} B \times_A T^2A & \xrightarrow{\nu_{TA}} & T(B \times_A TA) & \xrightarrow{T\nu} & T^2B \\ {\scriptstyle B\times_A \mu_A}\downarrow & & & & \downarrow{\scriptstyle \mu_B} \\ B \times_A TA & & \xrightarrow[\nu]{} & & TB \end{array} \qquad (\dagger)$$

where ν_{TA} is determined by its components $(\nu \cdot (B \times_A f), \pi')$ with respect to the pullback $T(B \times_A TA) = TB \times_{TA} T^2A$. Given that these diagrams commute in $\mathcal{E}$, it is not necessary to add an extra condition specifying that ν should be a map in $\mathcal{E}/A$; that this follows from the commutativity of the lower triangle of the right hand square above can be seen by consideration

of the diagram



In cases like this, we'll call the map ν an *unwiring* of the map f over the algebra (A, α).

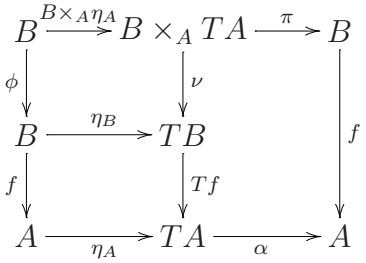
Example 2.2.1. If $\mathcal{E}$ has a terminal object 1, then 1 may be uniquely given the structure of a T-algebra as $(1, !_{T1})$. Then the notions of unwiring of $!_B$ over $(1, !_{T1})$ and unwiring of B coincide.

Intuitively, an unwiring of this kind may once more be thought of as a way to decompose things in B according to specified constructions made available by the monad T. However, the map f to A gives a little more structure; the ways of decomposing something in B are to correspond to ways that f of that thing might have been put together. Thus far, the notion as we have explained it corresponds to a map $\nu\colon B \times_A TA \to TB$ such that $Tf \cdot \nu = \pi'$. The additional conditions on ν above constrain ν to behave in a 'sensible' way.

There is another approach to the idea of a 'sensible' map ν in this context. ν is a factorisation of the pullback π' of f along α through Tf. One would expect to be able to use this to get, for example, a factorisation of the pullback of f along $\alpha \cdot T\alpha$ through $T^2 f$. Since (A, α) is a T-algebra, $\alpha \cdot T\alpha = \alpha \cdot \mu_A$, and ν will give a (potentially different) factorisation of the pullback of f along $\alpha \cdot \mu_A$ through $T^2 f$. For ν to be sensible, one would want these two factorisations to coincide, and would also want a similar condition with respect to the identity law on (A, α). We shall show that this notion of sensible map coincides with the notion of an unwiring of f introduced above.

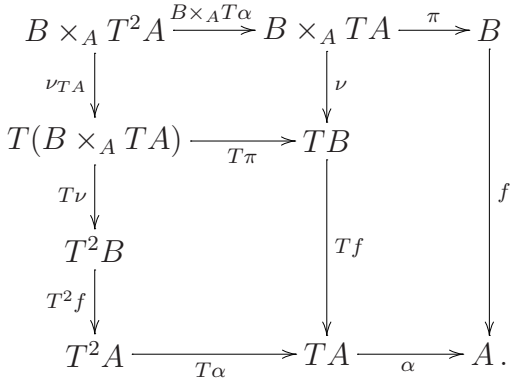
First, we'll treat the simpler case of the identity law $\alpha \cdot \eta_A = 1_A$. One can use ν to give a factorisation of the pullback of f along $\alpha \cdot \eta_A$ through f

as in the diagram

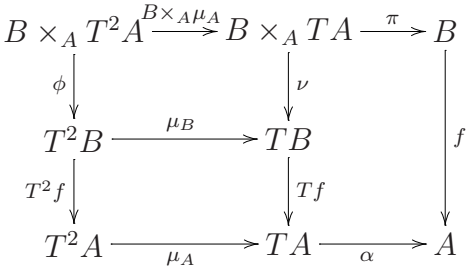


in which the desired factorisation ϕ is the unique map $B \to B$ such that $f \cdot \phi = f$ and $\eta_B \cdot \phi = \nu \cdot (B \times_A \eta_A)$. The condition we're interested in, $\phi = 1_B$, holds iff 1_B satisfies both of these equations. It trivially satisfies the first, and it satisfies the second iff the upper left triangle in (†) commutes.

The associativity law $\alpha \cdot T\alpha = \alpha \cdot \mu_A$ is a little harder. First, one can use ν to obtain the factorisation $T\nu \cdot \nu_{TA}$ of the pullback of f along $\alpha \cdot T\alpha$ through $T^2 f$ as in the diagram



Next, one can obtain a factorisation ϕ of the pullback of f along $\alpha \cdot \mu_A$ through $T^2 f$ as in the diagram



in which the desired factorisation ϕ is the unique map $B \times_A T^2A \to T^2B$ such that $T^2f \cdot \phi = \pi'$ and $\mu_B \cdot \phi = \nu \cdot (B \times_A \mu_A)$. The condition we're interested in, $\phi = T\nu \cdot \nu_{TA}$, holds iff $T\nu \cdot \nu_{TA}$ satisfies both of these equations. It satisfies the first of these equations by the definition of ν_{TA}, and it satisfies the second iff the right hand rectangle in (†) commutes.

Putting all of this together gives

PROPOSITION 2.2.2. *Let $f\colon B \to A$ be a map in $\mathcal{E}$, with (A, α) a T-algebra. Let ν be a factorisation of the pullback of f along α through Tf. ν is an unwiring of f over (A, α) iff the two factorisations of the pullback of f along $\alpha \cdot \nu_A = 1_A$ through f derived from ν coincide and the two factorisations of the pullback of f along $\alpha \cdot T\alpha = \alpha \cdot \mu_A$ through T^2f derived from ν coincide.*□

Some familiar concepts arise as special cases of the notion of an unwiring of a map.

EXAMPLE 2.2.3. If T is the identity monad on $\mathcal{E}$, then for any T-algebra (that is, object) A, T/A is the identity monad on $\mathcal{E}/A$ and so as in example 2.1.2 any map $B \xrightarrow{f} A$ has a unique unwiring over A, given by 1_B.

EXAMPLE 2.2.4. An unwiring over a monoid A of a map $f\colon B \to A$ with respect to the free monoid monad **fm** consists of a map ν which assigns to each element b of B and each factoring $\prod_{i\in[n]} a_i$ of $f(b)$ in A a sequence $(b_i)_{i\in[n]}$ such that $f(b_i) = a_i$ for all i, and which satisfies certain conditions. Partitioning B into sets $B_a = \{b \in B\colon f(b) = a\}$, ν gives maps $\nu_{\mathbf{a}}\colon B_{\prod_{i\in[n]} a_i} \to \prod_{i\in[n]} B_{a_i}$ for each sequence $\mathbf{a} = (a_i)_{i\in[n]}$ in $\mathbf{fm}A$. The conditions for ν to be an unwiring then state precisely that the maps $\nu_{\mathbf{a}}$ make $a \mapsto B_a$ a colax monoidal map $A \to \mathbf{Set}$. So the category $(\mathbf{fm}/A)$**-Arb** of $(\mathbf{fm}/A)$-arbeglas is equivalent to the category of such colax monoidal maps.

In particular, taking $A = 1$, **fm**-arbeglas correspond to colax monoidal maps from 1 to **Span**, that is, to comonoid objects in **Set**, which correspond in turn to sets; this is the content of example 2.1.3.

EXAMPLE 2.2.5. V is a strict map of monads from the free category monad **fc** on **Dig** to the identity monad on **Set**. For any category A, V lifts to a strict map of monads from $\mathbf{fc}/A$ to $1_{\mathbf{Set}}/VA$. So if ν is an unwiring of some map $f\colon B \to A$ of digraphs over A, then $V(\nu)$ is an unwiring of $V(f)$ with respect to 1. Thus as in example 2.1.2 $V(\nu)$ must be the identity map. As in example 2.2.4, $V(B)$ and $E(B)$ can be partitioned into sets $B_a = \{b \in V(B)\colon f(b) = a\}$ and $B_k = \{e \in E(B)\colon f(e) = k\}$. For each map $k\colon a \to a'$ in A, the source and target maps of B give B_k the structure of a span from B_a to $B_{a'}$. For each composable sequence $\mathbf{k} = (k_i)_{i\in[n]}$ of maps in A with composite k, ν induces a map $\nu_{\mathbf{k}}$ from the span B_k to the composite of the spans B_{k_i}. The conditions for ν to be an unwiring then say precisely that B_- gives a colax functor from A to the bicategory of spans. Thus the category of $\mathbf{fc}/A$-arbeglas is equivalent to the category of such colax functors.

PROPOSITION 2.2.6. *Let ν be an unwiring of a map $f\colon B \to A$, where (A, α) is a T-algebra. If ν is an isomorphism then B is a T-algebra and f is a map of T-algebras. This gives a bijection between unwirings of maps into A which are isomorphisms and maps $(B, \beta) \xrightarrow{f} (A, \alpha)$ of T-algebras such that*

$$\begin{array}{ccc} TB & \xrightarrow{Tf} & TA \\ {\scriptstyle\beta}\downarrow & & \downarrow{\scriptstyle\alpha} \\ B & \xrightarrow[f]{} & A \end{array}$$

is a pullback square.

Proof. This is proposition 2.1.10 applied to the monad T/A. □

In cases like this, we'll call the map f an *unwirable* map of T-algebras.

EXAMPLE 2.2.7. The identity map on any T-algebra is unwirable.

PROPOSITION 2.2.8. *Any composite of unwirable maps is unwirable.* □

Definition 2.2.9. *T-**Alg**$_u$ is the subcategory of T-**Alg** containing only the unwirable maps.*

Proposition 2.2.10. *Let $k\colon (A,\alpha) \to (A',\alpha')$ be an unwirable map of T-algebras. Let ν be an unwiring of some map $f\colon B \to A'$ over A'. Then $\langle \nu \cdot (B \times_{A'} Tk), Tk \cdot \pi' \rangle$ is an unwiring of $k^* f\colon B \times_{A'} A \to A$ over A.*

Proof. The pullback functor $k^*\colon \mathcal{E}/A' \to \mathcal{E}/A$ preserves all finite limits. Consideration of the diagram

$$\begin{array}{ccccc} T(B \times_{A'} A) & \longrightarrow & TA & \xrightarrow{\alpha} & A \\ \downarrow & & \downarrow{\scriptstyle Tk} & & \downarrow{\scriptstyle k} \\ TB & \xrightarrow[Tf]{} & TA' & \xrightarrow[\alpha']{} & A' \end{array}$$

shows that for any object $B \xrightarrow{f} A'$ of $\mathcal{E}/A'$ there is a canonical isomorphism $\psi_B\colon (T/A)k^* f \to k^*(T/A')f$. These canonical isomorphisms combine to give a natural transformation ψ, making (k^*, ψ) a weak map of monads from T/A' to T/A. Applying proposition 2.1.7 to this map gives the desired result. □

Corollary 2.2.11. *The injection T-**Alg**$_u \hookrightarrow T$-**Alg** creates pullbacks.* □

Example 2.2.12. Recall from example 2.2.4 that for any monoid A (**fm**$/A$)-arbeglas (f,ν) correspond to colax monoidal maps $A \to$ **Set**. If ν is an isomorphism, then so are all the components $\nu_{\mathbf{a}}$, so that the monoidal map is strong rather than just colax. Lax monoidal maps $A \to$ **Set** correspond to monoids with a map to A, and in this way if ν is an isomorphism there is a corresponding monoid structure on B.

Example 2.2.13. Recall from example 2.2.5 that for any category A (**fc**$/A$)-arbeglas (f,ν) correspond to colax functors from A to the bicategory of spans. If ν is an isomorphism then the functor involved is not just colax but also weak and so in particular lax. Lax functors from A to the bicategory of

spans correspond to categories with a functor to A, and in this way if ν is an isomorphism B has the structure of a category over A.

EXAMPLE 2.2.14. The functor $\Sigma\colon \mathbf{Set} \to \mathbf{Dig}$ sending a set X to the digraph with vertex set $\{*\}$ and edge set X is (the functor part of) a strict map of monads $\mathbf{fm} \to \mathbf{fc}$ (this map underlies the statement that a monoid is a 1-object category). For any monoid A, Σ lifts to a strict map of monads $\mathbf{fm}/A \to \mathbf{fc}/\Sigma A$. So as in definition 2.1.8 there is a functor Σ_* from $(\mathbf{fm}/A)$**-Arb** to $(\mathbf{fc}/\Sigma A)$**-Arb**. From example 2.1.3 we know that $(\mathbf{fm}/A)$**-Arb** is equivalent to the category of colax monoidal maps $A \to \mathbf{Set}$. So each $(\mathbf{fm}/A)$-arbegla B has a corresponding colax monoidal map, which can be considered as a colax functor $A \to \mathbf{Span}$ taking the unique object of A to the set $\{*\}$; this colax monoidal functor corresponds to the $(\mathbf{fc}/\Sigma A)$-arbegla $\Sigma_* B$.

2.3 Lifting exponentials via unwirings

A major use of unwirings of objects and maps is in the lifting of exponentials to categories of algebras. These liftings correspond to particular maps of monads; however, it is natural to begin by considering the mates of the maps of monads which will eventually be used. These mates can be discussed even in contexts where exponentials are not defined. They provide an alternative characterisation of unwirings in terms of distributive laws of monads.

Let T be a cartesian monad on a category $\mathcal{E}$ with all finite limits. Let ν be an unwiring of some object B, giving B the structure of a T-arbegla. Then for any object C of $\mathcal{E}$, pulling back ν along $T\pi\colon T(B \times C) \to TB$ gives

a diagram

$$\begin{array}{ccccc} B \times TC & \xrightarrow{l(\nu)_C} & T(B \times C) & \xrightarrow{T\pi'} & TC \\ {\scriptstyle B\times T!}\downarrow & & \downarrow{\scriptstyle T\pi} & & \downarrow{\scriptstyle T!} \\ B \times T1 & \xrightarrow[\nu]{} & TB & \xrightarrow[T!]{} & T1\,. \end{array}$$

Since the maps ν_C are formed in this way by pullback, they collectively form a cartesian natural transformation $l(\nu)\colon B \times T- \to T(B \times -)$.

Proposition 2.3.1. *For any unwiring ν of B as above, $l(\nu)$ is a distributive law of the comonad $B \times -$ over T. This gives a bijection between unwirings ν of B and cartesian distributive laws of $B \times -$ over T.*

Proof. The commutativity of the diagrams

$$\begin{array}{ccc} B \times C & \xrightarrow{B\times\nu_C} & B \times TC \\ {\scriptstyle \nu_{B\times C}}\downarrow & \swarrow{\scriptstyle l(\nu)_C} & \\ T(B \times C) & & \end{array} \qquad \text{and} \qquad \begin{array}{ccccc} B \times T^2C & \xrightarrow{l(\nu)_{TC}} & T(B \times TC) & \xrightarrow{Tl(\nu)_C} & T^2(B \times C) \\ {\scriptstyle B\times\mu_C}\downarrow & & & & \downarrow{\scriptstyle \mu_{B\times C}} \\ B \times TC & & \xrightarrow[\nu_C]{} & & T(B \times C) \end{array}$$

is obtained by pulling back two of the diagrams in the definition of an unwiring along $T\pi\colon T(B \times C) \to TB$.

The commutativity of the diagram

$$\begin{array}{ccc} B \times TC & \xrightarrow{\pi'} & TC \\ {\scriptstyle l(\nu)_C}\downarrow & \nearrow{\scriptstyle \pi'} & \\ T(B \times C) & & \end{array}$$

follows from the fact that the outer rectangle in the diagram used to define $l(\nu)_C$ is the pullback

$$\begin{array}{ccc} B \times TC & \xrightarrow{\pi'} & TC \\ {\scriptstyle B\times T!}\downarrow & & \downarrow{\scriptstyle T!} \\ B \times T1 & \xrightarrow[\pi']{} & T1\,. \end{array}$$

Finally, the commutativity of the diagram

$$\begin{array}{ccccc} B \times TC & & \xrightarrow{\quad l(\nu_C) \quad} & & T(B \times C) \\ {\scriptstyle \delta_B \times TC}\downarrow & & & & \downarrow{\scriptstyle T(\delta_B \times C)} \\ B \times B \times TC & \xrightarrow[B \times l(\nu)_C]{} & B \times T(B \times C) & \xrightarrow[l(\nu)_{B \times C}]{} & T(B \times B \times C) \end{array}$$

can be established by composing with the projection maps from the bottom right hand corner to its two components, considered as the pullback $TB \times_{T1} T(B \times C)$.

The inverse operation n to l sends a distributive law λ to its component at 1. Trivially $n(l(\nu)) = \nu$ for any unwiring ν of B. Conversely, let λ be any cartesian distributive law of $B \times -$ over T. Then for each C, $T\pi \cdot l(n(\lambda))_C = n(\lambda) \cdot B \times T! = T\pi \cdot \lambda_C$, and $T\pi' \cdot l(n(\lambda))_C = \pi' = T\pi' \cdot \lambda_C$. Since $T(B \times C)$ is a pullback with projections $T\pi$ and $T\pi'$, $l(n(\lambda))_C = \lambda_C$. Thus $l(n(\lambda)) = \lambda$. □

Suppose we have some arbegla (B, ν), with corresponding distributive law λ. It is natural to consider the double Kleisli category $\mathbf{Kl}(\lambda)$. The objects are the same as those of $\mathcal{E}$. The maps $C \to D$ in $\mathbf{Kl}(\lambda)$ are given by maps $B \times C \to TD$ in $\mathcal{E}$, and the composition can be determined in terms of the distributive law λ in the usual way. Intuitively, a map from C to D assigns to every pair of similar items from B and C a composable collection of items from D. The item from B can be decomposed over the shape of this collection using the unwiring, to give a composable collection of pairs of similar items from B and D. Given a map in the Kleisli category from D to E, we can map this to a composable collection of composable collections in E, which can be pasted together to give a final composable collection in E.

For the remainder of this section, we shall make the additional assumption that $\mathcal{E}$ is cartesian closed. Thus each map $B \times -$ has a right adjoint $-^B$. For any unwiring ν of B, $l(\nu)$ gives $B \times -$ the structure of a cartesian colax map of monads from T to itself. Let $m(\nu)$ be the mate of $l(\nu)$ with

respect to the adjuction $(B \times - \dashv -^B)$. Then $(-^B, m(\nu))$ is a lax map of monads from T to itself. Thus the functor $-^B$ lifts to an endomorphism of the category of T-algebras. In particular, it follows that any unwirable T-algebra is exponentiable.

Since $-^B$ is right adjoint to the comonad $B \times -$, it inherits a monad structure, in such a way that $m(\nu)$ becomes a distributive law of T over $-^B$. So the lifted endomorphism $-^B$ on the category of T-algebras also inherits a monad structure. Similarly, the monad T extends to the Kleisli category for $-^B$. The Kleisli category for this extended monad is the double Kleisli category $\mathbf{Kl}(n(\nu))$ introduced above. Similarly, the distributive law $m(\nu)$ gives $(-^B) \cdot T$ the structure of a monad, and the Kleisli category for this monad is also $\mathbf{Kl}(n(\nu))$.

Remark 2.3.2. *This explains why unwirable objects were so rare in many of the cases considered so far. It is known that for many simple monads (even on cartesian closed categories) there are very few exponentiable objects in the category of algebras. For example, the terminal monoid is the only exponentiable monoid, and so we could not have expected to find any more unwirable monoids. However, in the next chapter we'll see that, for many free multicategory monads, there are richly structured unwirable algebras (that is, unwirable multicategories).*

It is natural to ask at this point whether the classes of exponentiable and unwirable algebras for a monad correspond. That they do not can be seen by examining the free category monad: every category is exponentiable, but only discrete categories are unwirable. The unwirable algebras form a well-behaved subclass of the exponentiable algebras, and this good behaviour will be exploited in the constructions of the next chapter.

Example 2.3.3. Recall from example 2.1.3 that **fm**-arbeglas correspond to sets. So the statement above reduces in this case to the claim that for any set X the functor $-^X$ lifts to the category of monoids (pointwise evaluation).

EXAMPLE 2.3.4. Recall from example 2.1.9 that **fc**-arbeglas correspond to indexed families $(B_x)_{x\in X}$ of sets. The corresponding unwiring has underlying digraph with vertex set X and edge set $\bigsqcup_{x\in X} B_x$, with each edge in B_x having source and target at x. For any other digraph C, C^B has as its vertex set the set of functions from X to $V(C)$. An edge from f to g in C^B is given by a family $(k_x)_{x\in X}$ of functions with $k_x\colon B_x \to C(f(x), g(x))$, where $C(a,b)$ is the set of edges in C from a to b. The statement above reduces to the fact that if C is a category then so is C^B, with the operations computed pointwise.

For the rest of this section, we'll impose the even stronger condition that the category $\mathcal{E}$ is locally cartesian closed (that is, $\mathcal{E}$ has pullbacks and each slice category of $\mathcal{E}$ is cartesian closed). Let T be a cartesian monad on $\mathcal{E}$, and let (A,α) be a T-algebra. Let $f\colon B \to A$ be an object of $\mathcal{E}/A$, and ν an unwiring of f over A. Then the operation $-^f$ lifts to an endomorphism of the category (T/A)**-Alg** $= T$**-Alg**$/(A,\alpha)$ of T-algebras over A.

EXAMPLE 2.3.5. An $\mathbb{N}_0$-*graded monoid* is a monoid A together with a monoid homomorphism $l\colon A \to \mathbb{N}_0$. For example, any free monoid is $\mathbb{N}_0$-graded, with the grading given by length. $\mathbb{N}_0$-graded monoids are algebras for the monad $\mathbf{fm}/\mathbb{N}_0$ on $\mathbf{Set}/\mathbb{N}_0$. A $(\mathbf{fm}/\mathbb{N}_0)$-arbegla is given by a colax functor $\mathbb{N}_0 \to \mathbf{Set}$. Such a functor is given by a sequence $(X_n)_{n\in\mathbb{N}_0}$ of sets together with maps $\nu_{\mathbf{n}}\colon X_{\sum_{i\in[m]} n_i} \to \prod_{i\in[m]} X_{n_i}$ for each sequence $\mathbf{n} = (n_i)_{i\in[m]}$ of natural numbers, satisfying some laws. The exponentiation functor this induces on the category $\mathbf{Set}/\mathbb{N}_0$ of $\mathbb{N}_0$-graded sets sends the graded set $(A_n)_{n\in\mathbb{N}_0}$ to the graded set $A^X = (A_n^{X_n})_{n\in\mathbb{N}_0}$. If A is a graded monoid, then this yields a monoid operation on A^X with the product of the sequence $(k_i\colon X_{n_i} \to A_{n_i})_{i\in[m]}$ given by $k\colon X_{\sum_{i\in[m]} n_i} \to A_{\sum_{i\in[m]} n_i}; x \mapsto \prod_{i\in[m]} k_{n_i}(\pi_i(\nu_{\mathbf{n}}(x)))$.

EXAMPLE 2.3.6. Let A be an $\mathbb{N}_0$-graded monoid. Then there is a corresponding $\mathbb{N}_0$-graded monoid A_k^+ with the elements of grade at most k given

by those of A, and with a unique element $*_n$ of each grade greater than k. This construction is given by the lifting in the last example, taking

$$X_i = \begin{cases} \{*\} & \text{if } i \leq k \\ \emptyset & \text{if } i > k \end{cases}$$

(the maps $\nu_{\mathbf{n}}$ are then uniquely determined).

Chapter 3

Unwirings of multicategories

As we discussed in the introduction, the mathematical objects taken to be games in a category of games normally already have a combinatorial structure of some kind, so that they form a category as combinatorial objects of that kind. For example, trees, digraphs and the recursive structures Conway worked with all form categories of this kind. As well as the maps as combinatorial objects, there is a second kind of map between these games, given by strategies, for which the composition is more subtle.

This is the reason for thinking that some 2-dimensional structure may be appropriate for the study of games, and we would suggest that the proper 2-dimensional structures to use are **fc**-multicategories and a specialised kind of unwiring of **fc**-graphs, called playpens. The objects of the **fc**-multicategories and playpens in question will be games. The vertical maps, which have a straightforward composition, correspond to the maps-as-combinatorial-objects of the games. The horizontal maps correspond to plays (in the playpens) or strategies (in the **fc**-multicategories). Prima facie, the strategies obtained this way do not compose. However, the 2-cells will contain enough information to control composition. More precisely, the **fc**-multicategories in question will be representable. They will therefore inherit the structure of

double categories, giving a composition operation on the strategies.

In this chapter, we shall develop the theory of the chapter 2 in the context of **fc**-multicategories to the point where we can make the comments above precise for the example of digraph games. We won't, however, discuss representability yet. That will be left for chapter 4.

In section 3.1 we'll discuss unwirings in the context of T-multicategories. Unwirable maps of suitable T-multicategories turn out to have a simple characterisation. We'll also explain the first really interesting example of unwirings in this document - unwirings of plain multicategories. We'll focus on the special case of playpens in 3.2, and outline some of their basic properties and constructions. Then in section 3.3 we'll introduce in some detail two particular playpens, MAT and RING, which will control the framework for plays to count as horizontal arrows and the combinatorics of composition of strategies for digraph games respectively. Finally in section 3.4 we'll show how combining the constructions so far yields an **fc**-multicategory of digraph games, and we'll outline how the construction from there of a category of digraph games will proceed.

3.1 Unwirings of maps of multicategories

We shall be interested in unwirings of maps with respect to two monads in particular; the free plain multicategory monad, and the free **fc**-multicategory monad. The first of these monads is explained in [13, §2.3], and the second is a special case of the more general construction in §6.5 of that book. Both of these kinds of multicategory were introduced in the 'Primer on multicategories' at the start of this document. We will use the notation introduced in that primer. Much of the theory we will rely on can be developed in the more general setting outlined there.

Recall that for T a suitable[1] monad on a suitable category $\mathcal{E}$, the forgetful functor $(T,\mathcal{E})$-**Mult** $\to$ $(T,\mathcal{E})$-**Gph** sending any T-multicategory to its underlying T-graph is monadic. So there is a corresponding monad T^+ on $\mathcal{E}^+ = (T,\mathcal{E})$-**Gph**, called the free T-multicategory monad, algebras for which are T-multicategories.

EXAMPLE 3.1.1. $1^+_{\mathbf{Set}}$ is **fc**, the free category monad. $\mathbf{fm}^+$ is the free plain multicategory monad.

The details of the standard construction of T^+ are a little technical, and we shall wish to refer to these details later on, so we'll sketch them out at this point. Let (A,α) be a T^+ algebra, that is, a T-multicategory. In an appendix to [13], T^+A is constructed as a colimit of a diagram

$$A^{(0)} \longrightarrow A^{(1)} \longrightarrow A^{(2)} \longrightarrow \cdots$$

in which the objects $A^{(i)}$ are defined inductively by $A^{(0)} = A_0$ and $A^{(n+1)} = A_0 + A_1 \circ A^{(n)}$. α induces the identity-assignment map $\alpha_0 \colon A_0 \to A$ and the multiplication map $\alpha_2 \colon A \circ A \to A$ by composition with the injections $i_A \colon A_0 \to T^+A$ and $m_A \colon A \circ A \to T^+A$ (these injections are the components at A of cartesian natural transformations to T^+). Conversely, given maps α_0 and α_2 satisfying associativity and identity laws, it is possible to recursively define maps $\alpha^{(i)} \colon A^{(i)} \to A$ which paste together to give a map α giving A the structure of a T^+-algebra. This gives the correspondence between T^+-algebras and T-multicategories.

Now let $(B \xrightarrow{f} A, \nu)$ be a (T^+/A)-arbegla. We might hope to be able to represent ν in terms of maps ν_0 and ν_2 making the diagrams

$$\begin{array}{ccc} B \times_A A_0 & \xrightarrow{\nu_0} & B_0 \\ & {\scriptstyle \pi'} \searrow & \downarrow {\scriptstyle f_0} \\ & & A_0 \end{array} \qquad \text{and} \qquad \begin{array}{ccc} B \times_A (A_1 \circ A_1) & \xrightarrow{\nu_2} & B_1 \circ B_1 \\ & {\scriptstyle \pi'} \searrow & \downarrow {\scriptstyle f_1 * f_1} \\ & & A_1 \circ A_1 \end{array}$$

[1] 'Suitable' here is a technical term, defined in [13].

commute. Indeed, it will turn out to be possible to give an alternate definition of unwirings with respect to T^+ in terms of such pairs of maps, satisfying appropriate relations. A definition reminiscent of that at the start of section 2.3 gives a convenient notation for expressing these relations.

DEFINITION 3.1.2. *Let $\nu_2 \colon B \times_A (A_1 \circ A_1) \to B_1 \circ B_1$ be as above, and let $c \colon C \to A$ and $d \colon D \to A$ be maps of T-graphs, which are identities on objects (so $c_0 = d_0 = 1_{A_0}$). The lifting $l(\nu_2)_{c,d}$ of ν_2 along c and d is given by the pullback*

$$\begin{array}{ccccc}
B \times_A (C_1 \circ D_1) & \xrightarrow{l(\nu_2)_{c,d}} & (B \times_A C) \circ (B \times_A D) & \xrightarrow{(f \times_A C) * (f \times_A D)} & C \circ D \\
\downarrow{\scriptstyle B \times_A (c*d)} & & \downarrow{\scriptstyle (B \times_A c) * (B \times_A d)} & & \downarrow{\scriptstyle c*d} \\
B \times_A (A \circ A) & \xrightarrow[\nu_2]{} & B \circ B & \xrightarrow[f*f]{} & A \circ A \,.
\end{array}$$

DEFINITION 3.1.3. *A* weak unwiring *of a map f of T-multicategories is a pair of maps $\nu_0 \colon B \times_A A_0 \to B_0$ and $\nu_2 \colon B \times_A (A_1 \circ A_1) \to B_1 \circ B_1$ making the following diagrams commute:*

$$\begin{array}{ccc}
B \times_A A_0 & \xrightarrow{\nu_0} & B_0 \\
& {\scriptstyle \pi'} \searrow & \downarrow{\scriptstyle f_0} \\
& & A_0
\end{array}
\qquad\qquad
\begin{array}{ccc}
B \times_A (A_1 \circ A_1) & \xrightarrow{\nu_2} & B_1 \circ B_1 \\
& {\scriptstyle \pi'} \searrow & \downarrow{\scriptstyle f_1 * f_1} \\
& & A_1 \circ A_1
\end{array}$$

$$\begin{array}{ccc}
B \times_A (A_1 \circ A_0) & \xrightarrow{l(\nu_2)_{1_A,\mathrm{ids}}} & (B \times_A A) \circ (B \times_A A_0) \\
\downarrow{\scriptstyle \wr} & & \downarrow{\scriptstyle 1 * \nu_0} \\
B & \xrightarrow{\sim} & (B \times_A A) \circ B_0
\end{array}$$

$$\begin{array}{ccc}
B \times_A (A_0 \circ A_1) & \xrightarrow{l(\nu_2)_{\mathrm{ids},1_A}} & (B \times_A A_0) \circ (B \times_A A) \\
\downarrow{\scriptstyle \wr} & & \downarrow{\scriptstyle \nu_0 * 1} \\
B & \xrightarrow{\sim} & B_0 \circ (B \times_A A)
\end{array}$$

$$\begin{array}{ccc} B \times_A (A_1 \circ A_1 \circ A_1) & \xrightarrow{l(\nu_2)_{1_A,\mathrm{comp}}} & (B \times_A A_1) \circ (B \times_A (A_1 \circ A_1)) \\ \downarrow{\scriptstyle l(\nu_2)_{\mathrm{comp},1_A}} & & \downarrow{\scriptstyle 1*\nu_2} \\ (B \times_A (A_1 \circ A_1)) \circ (B \times_A A_1) & \xrightarrow[\nu_2 * 1]{} & B_1 \circ B_1 \circ B_1 \end{array}$$

Proposition 3.1.4. *Let $\nu\colon B \times T^+A \to T^+B$ be an unwiring of the map f from the T-graph B to the T-multicategory A. Let ν_0 be given by its components $(\nu \cdot (B \times_A i_A), \pi')$ with respect to the pullback $B_0 \cong T^+B \times_{T^+A} A_0$. Let ν_2 be given by its components $(\nu \cdot (B \times_A m_A), \pi')$ with respect to the pullback $B_1 \circ B_1 \cong T^+B \times_{T^+A} (A_1 \circ A_1)$. Then (ν_0, ν_2) is a weak unwiring of f.*

Proof. The commutativity of the first two diagrams is immediate from the definition of ν_0 and ν_2. The commutativity of the last diagram can be deduced by composing the maps on each side with the projections of the pullback $B \circ B \circ B \cong T^+B \times_{T^+A} (A \circ A \circ A)$. The composites with the first projection are equal by the commutativity of two similar diagrams, the first of which is

$$\begin{array}{ccccc} B \times_A (A \circ A \circ A) & \xrightarrow{l(\nu_2)_{1_A,\mathrm{comp}}} & (B \times_A A) \circ (B \times_A (A \circ A)) & \xrightarrow{1*\nu_2} & B \circ B \circ B \\ \downarrow & & \downarrow & & \downarrow \\ B \times_A (T^+)^2 A & \xrightarrow{\nu_{T^+1}} & T^+(B \times_A T^+A) & \xrightarrow{T^+\nu} & (T^+)^2 B \\ \downarrow & & & & \downarrow \\ B \times_A T^+A & & \xrightarrow[\nu]{} & & T^+B\,. \end{array}$$

The composites with the second projection each evaluate to π'. A similar argument applies to each of the middle two diagrams, though now the projections in question are those for the pullback $B \cong T^+B \times_{T^+A} A$. □

This process is invertible; given a weak unwiring (ν_0, ν_2), there is a unique unwiring from which it arises in this way. This unwiring $\nu\colon B \times_A T^+A \to T^+B$ is constructed from the components $\nu^{(n)}\colon B \times_A A^{(n)} \to B^{(n)}$. These components in turn are constructed recursively by $\nu^{(0)} = \nu_0$ and

$$\nu^{(n+1)} = \nu_0 + (1_B * \nu^{(n)}) \cdot l(\nu_2)_{1_A,\mu_A^{(n)}}\,.$$

PROPOSITION 3.1.5. *Let ν be an unwiring of a map f from an* **fc***-graph B to an* **fc***-multicategory A. Then the maps $s \cdot \pi$ and $\eta_B \cdot t \cdot \pi$ from $B \times_A A_0$ to TB_0 are equal.*

Proof. Using the definition above, each of these may easily be shown to equal the symmetrical composite

$$\begin{array}{ccccc} B \times_A A_0 & \xrightarrow{\sim} & B \times_A (A_0 \circ A_0) & \xrightarrow{\nu_{\mathrm{ids},\mathrm{ids}}} & (B \times_A A_0) \circ (B \times_A A_0) \\ & & & & \downarrow{\scriptstyle \gamma} \\ TB_0 & \xleftarrow[s]{} & TB & \xleftarrow[T\pi]{} & T(B \times_A A_0) \end{array}$$

in which the map γ is the first projection of the pullback structure $T(B \times_A A_0) \times_{TA_0} (B \times_A A_0)$ on $(B \times_A A_0) \circ (B \times_A A_0)$. □

PROPOSITION 3.1.6. *A map f of T-multicategories is unwirable iff the square*

$$\begin{array}{ccc} B_1 \circ B_1 & \xrightarrow{\mathrm{comp}} & B_1 \\ {\scriptstyle f_1 * f_1}\downarrow & & \downarrow{\scriptstyle f_1} \\ A_1 \circ A_1 & \xrightarrow[\mathrm{comp}]{} & A_1 \end{array}$$

is a pullback.

Proof. f is unwirable iff the square

$$\begin{array}{ccc} T^+B & \xrightarrow{\mathrm{comp}} & B \\ {\scriptstyle T^+f}\downarrow & & \downarrow{\scriptstyle f} \\ T^+A & \xrightarrow[\mathrm{comp}]{} & A \end{array}$$

is a pullback. However, T^+A is generated by $A^{(0)}$ and $A^{(2)}$ and a similar fact holds for B, so since the injections i and m are cartesian and pullbacks commute with colimits of shape $\mathbb{N}_0$ this reduces to the condition that the squares

$$\begin{array}{ccc} B_0 & \xrightarrow{\mathrm{ids}} & B_1 \\ {\scriptstyle f_0}\downarrow & & \downarrow{\scriptstyle f_1} \\ A_0 & \xrightarrow[\mathrm{ids}]{} & A_1 \end{array} \quad \text{and} \quad \begin{array}{ccc} B_1 \circ B_1 & \xrightarrow{\mathrm{comp}} & B_1 \\ {\scriptstyle f_1 * f_1}\downarrow & & \downarrow{\scriptstyle f_1} \\ A_1 \circ A_1 & \xrightarrow[\mathrm{comp}]{} & A_1 \end{array}$$

are both pullbacks.

It is therefore sufficient to show that the first of these conditions follows from the second. Form the pullback

$$\begin{array}{ccc} P & \xrightarrow{u} & B_1 \\ {\scriptstyle v}\downarrow & & \downarrow{\scriptstyle f_1} \\ A_0 & \xrightarrow[\text{ids}]{} & A_1\,. \end{array}$$

By the second condition, this rectangle may be factored as

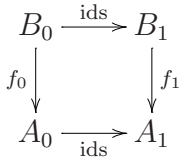
$$\begin{array}{ccccc} P & \xrightarrow{w} & B_1 \circ B_1 & \xrightarrow{\text{comp}} & B_1 \\ {\scriptstyle v}\downarrow & & \downarrow{\scriptstyle f_1 * f_1} & & \downarrow{\scriptstyle f_1} \\ A_0 & \xrightarrow[(\eta\cdot\text{ids},\text{ids})]{} & A_1 \circ A_1 & \xrightarrow[\text{comp}]{} & A_1\,. \end{array}$$

Now consider the diagram

$$\begin{array}{ccccccc} & & & \overset{1_{B_1}}{\frown} & & & \\ P & \xrightarrow{u} & B_1 & \xrightarrow[(\eta,\text{ids}\cdot t)]{} & B_1 \circ B_1 & \xrightarrow{\text{comp}} & B_1 \\ {\scriptstyle v}\downarrow & & {\scriptstyle f_1}\downarrow & & \downarrow{\scriptstyle f_1 * f_1} & & \downarrow{\scriptstyle f_1} \\ A_0 & \xrightarrow[\text{ids}]{} & A_1 & \xrightarrow{(\eta,\text{ids}\cdot t)} & A_1 \circ A_1 & \xrightarrow[\text{comp}]{} & A_1\,. \\ & & & \underset{1_{A_1}}{\smile} & & & \end{array}$$

Since the first square and the composite of the second and third squares are both pullbacks, the whole rectangle is a pullback. Thus the composite of the first and second squares must be the first square of the previous diagram, and in particular $w = (\eta, \text{ids}\cdot t)\cdot u$. An analogous argument shows that also $w = (\eta\cdot\text{ids}\cdot s, 1)\cdot u$. Thus $u = \text{ids}\cdot t\cdot u$. We are now a position to show that $t\cdot u$ is inverse to $(\text{ids}, f_0)\colon B_0 \to P$. For $t\cdot u\cdot(\text{ids}, f_0) = t\cdot\text{ids} = 1_{B_0}$ and $(\text{ids}, f_0)\cdot t\cdot u = (\text{ids}\cdot t\cdot u, f_0\cdot t\cdot u) = (u, t\cdot f_1\cdot u) = (u, t\cdot\text{ids}\cdot v) = (u, v) = 1_P$.

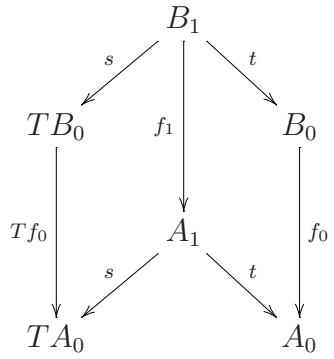
Thus



is a pullback, as required. □

Using the definition of weak unwiring, it is possible to define the notions of unwirable T-multicategory and unwiring of a map of T-graphs even if the category $\mathcal{E}$ and the monad T are not required to be suitable, and propositions 3.1.5 and 3.1.6 still hold. However, since we will not need to make use of such examples, we will not explore this aspect of the theory any further here.

Recall that for any map $f\colon B \to A$ of T-graphs, the diagram



must commute. f is a *discrete opfibration* iff the left-hand square in this diagram is a pullback. Discrete opfibrations give us examples of unwirable T-multicategories.

PROPOSITION 3.1.7. *Let $f\colon B \to A$ be a discrete opfibration of a T-multicategory A. Then the T-graph B inherits a unique T-multicategory structure making f an unwirable map of T-multicategories.*

Proof. We shall rely on proposition 3.1.6. Both squares in the rectangle

$$\begin{array}{ccccc} B_1 \circ B_1 & \longrightarrow & B_1 & \xrightarrow{f_1} & A_1 \\ \downarrow & & \downarrow{\scriptstyle s} & & \downarrow{\scriptstyle s} \\ TB_1 & \xrightarrow[Tt]{} & TB_0 & \xrightarrow[Tf_0]{} & TA_0 \end{array}$$

are pullbacks and so the rectangle itself is also a pullback. This rectangle can also be decomposed as

$$\begin{array}{ccccc} B_1 \circ B_1 & \xrightarrow{f*f} & A_1 \circ A_1 & \longrightarrow & A_1 \\ \downarrow & & \downarrow & & \downarrow{\scriptstyle s} \\ TB_1 & \xrightarrow[Tf_1]{} & TA_1 & \xrightarrow[Tt]{} & TA_0 \end{array}$$

and since in this decomposition the right hand square is a pullback, so is the left hand square. Thus all three squares in the rectangle

$$\begin{array}{ccccccc} B_1 \circ B_1 & \longrightarrow & TB_1 & \xrightarrow{Ts} & T^2B_0 & \xrightarrow{\mu} & TB_0 \\ {\scriptstyle f*f}\downarrow & & {\scriptstyle Tf_1}\downarrow & & {\scriptstyle T^2f_0}\downarrow & & \downarrow{\scriptstyle Tf_0} \\ A_1 \circ A_1 & \longrightarrow & TA_1 & \xrightarrow[Ts]{} & T^2A_0 & \xrightarrow[\mu]{} & TA_0 \end{array}$$

are pullbacks, and so the rectangle itself is. The bottom arrow in this rectangle may also be factored as $s \cdot \mathrm{comp}$ and so the rectangle itself factors as

$$\begin{array}{ccccc} B_1 \circ B_1 & \longrightarrow & B_1 & \xrightarrow{s} & TB_0 \\ {\scriptstyle f*f}\downarrow & & {\scriptstyle f_1}\downarrow & & \downarrow{\scriptstyle Tf_0} \\ A_1 \circ A_1 & \xrightarrow[\mathrm{comp}]{} & A_1 & \xrightarrow[s]{} & TA_0 \end{array}$$

since the right hand square is a pullback. This defines a map $\mathrm{comp}\colon B_1 \circ B_1 \to B_1$, and a similar definition yields a map $\mathrm{ids}\colon B_0 \to B_1$. These maps give B the structure of a T-multicategory, and since the square

$$\begin{array}{ccc} B_1 \circ B_1 & \xrightarrow{\mathrm{comp}} & B_1 \\ {\scriptstyle f_1*f_1}\downarrow & & \downarrow{\scriptstyle f_1} \\ A_1 \circ A_1 & \xrightarrow[\mathrm{comp}]{} & A_1 \end{array}$$

is a pullback, by proposition 3.1.6 they also give f the structure of an unwirable map of T-multicategories from B to A. □

PROPOSITION 3.1.8. *Suppose A is a T-multicategory and $f\colon B \to A$ is a discrete fibration. Then the T-graph B inherits a unique T-multicategory structure making f an unwirable map of T-multicategories.*

Proof. The argument is similar to that for proposition 3.1.7, so we shall omit it. □

The last two propositions are occasionally useful, since discrete fibrations and opfibrations are preserved by the change-of-shape functors defined in [13, §6.7], whereas unwirings in general are not. However, we are not currently aware of a simple characterisation of those unwirings which are preserved by such functors, or even of a general class of such unwirings which incorporates the cases covered in propositions 3.1.7 and 3.1.8.

Finally, it is necessary to establish some notation for dealing with unwirable plain multicategories and **fc**-multicategories.

DEFINITION 3.1.9. *Let $f\colon C \to A$ be any unwirable map of plain multicategories. Then for each pair (c, s), where s is a composable collection $(a, (a_i)_{i\in[n]})$ in $A \circ A$ and c is a cell in C of the same arity, there is a unique composable collection $(c_s, (c_{s,i})_{i\in[n]})$ in $C_1 \circ C_1$ of cells, of shape s (that is, such that $f(c_s) = a_s$) and $f(c_{s,i}) = a_i$) such that $c_s \cdot (c_{s,i})_{i\in[n]} = c$. We'll call this composable collection the* decomposition *of c over s.*

EXAMPLE 3.1.10. The terminal object 1 of the category **fm-Gph** has a single vertex and a single n-ary edge for each n. We'll call the unique n-ary edge n. 1 has the structure of a plain multicategory ($\mathbf{fm}^+$-algebra) in a unique way, with structure map !. Here the composition is given by $n \cdot (n_i)_{i\in[n]} = \sum_{i\in n} n_i$. Let C be any unwirable plain multicategory. Then the map $!\colon C \to 1$ is unwirable.

Since the square

$$\begin{array}{ccc} C_0 & \xrightarrow{\text{ids}} & B_1 \\ {\scriptstyle !_0}\downarrow & & \downarrow{\scriptstyle !_1} \\ 1_0 & \xrightarrow[\text{ids}]{} & 1_1 \end{array}$$

is a pullback, any cell c of arity 1 in C_1, is an identity cell; the underlying category $U(C)$ of 1-ary cells in C is discrete. So if the collection of n-ary cells is considered as a profunctor $P_n \colon U(C) \to U(C)^n$, then, for any composable collection $s = (n, (n_i)_{i\in[n]})$ in $1 \circ 1$, $(\prod_{i\in[n]} P_{n_i}) \cdot P_n$ is the set of composable collections in $C_1 \circ C_1$ of shape s, which is isomorphic to $P_{\sum_{i\in[n]} n_i}$ and so to $(\prod_{i\in[m]} P_{m_i}) \cdot P_m$ for any other finite sequence $(m_i)_{i\in[m]}$ with the same sum. These isomorphisms give C_0 the structure of a discrete promonoidal category, and working back it is clear that the underlying multicategory of any discrete promonoidal category is unwirable. This correspondence extends to an equivalence from the category of unwirable plain multicategories (that is, the category of $\mathbf{fm}^+$-arbeglas) to the category of discrete promonoidal categories.

Example 3.1.10 gives an interesting case of the exponentiability construction of section 2.3. The category of **fm**-graphs is cartesian closed. Given two **fm**-graphs A and C, the vertices of A^C are functions from the objects of C to the objects of A. The edges of A^C from $(f_i)_{i\in[n]}$ to f are given by functions k from the edges of C to the edges of A sending edges with source $(c_i)_{i\in[n]}$ and target c to edges with source $(f_i(c_i))_{i\in[n]}$ and target $f(c)$. If C is unwirable and A is a plain multicategory, then A^C is also a plain multicategory. Given a composable collection $(k, (k_i)_{i\in[n]})$ of cells in A^C, where k_i has arity n_i, the composite of this collection is the function sending an edge c of C to $k(c_s) \cdot (k_i(c_{s,i}))_{i\in[n]}$, where s is $(n, (n_i)_{i\in[n]})$ and $(c_s, (c_{s,i})_{i\in[n]})$ is the decomposition of c over s.

Set is a monoidal category, and so it can also be considered as a plain mul-

ticategory. The maps from $(X_i)_{i\in[n]}$ to X are given by functions $\prod_{i\in[n]} X_i \to x$. Specialising the previous example to the case $A = \mathbf{Set}$, there is for any unwirable plain multicategory C a corresponding plain multicategory whose objects are functions from A_0 to **Set**. The discrete promonoidal category corresponding to a plain multicategory C has the discrete category on C_0 as its underlying category, and the structural profunctors P_n given by the collections of maps of arity n. As in section 1.2, there is a monoidal structure on the category of presheaves from C_0 to **Set**, which is just the set of functions from C_0 to **Set**, since C_0 is discrete. It is easy to check that the underlying multicategory of this monoidal structure is the multicategory $\mathbf{Set}^C$ constructed above. So in this case the multicategory A^C has some additional structure; it is a monoidal category.

Definition 3.1.11. *Let $f\colon C \to A$ be any unwirable map of* **fc***-multicategories. Then for each pair (k, p), where k is a vertical map in C and $p = (l, l')$ is a composable pair of vertical maps in A with $l \cdot l' = f(k)$, there is a unique composable pair (p_k, p'_k) of vertical maps in C with $f(p_k) = l$, $f(p'_k) = l'$ and $p_k \cdot p'_k = k$. We'll call this composable pair the* decomposition *of k over p.*

For each pair (c, s), where s is a composable collection $(a, (a_i)_{i\in[n]})$ in $A \circ A$ and c is a cell in C of the same arity, there is a unique composable collection $(c_s, (c_{s,i})_{i\in[n]})$ in $C_1 \circ C_1$ of cells, of shape s (that is, such that $f(c_s) = a_s$) and $f(c_{s,i}) = a_i$) such that $c_s \cdot (c_{s,i})_{i\in[n]} = c$. We'll call this composable collection the decomposition *of c over s.*

3.2 Playpens

For the constructions we'll be doing later, we'll be making a lot of use of a particular kind of unwiring of maps of **fc**-graphs, which we'll call playpens. A playpen is an **fc**-graph whose vertical structure already forms a category and which can be unwired over this vertical structure. To make this more

precise, we need some notation for dealing with the vertical structures of **fc**-graphs and **fc**-multicategories.

DEFINITION 3.2.1. *For any set X, the* indiscrete *digraph $I(X)$ on X has X as its set of vertices and a unique edge between any pair of vertices.*

The map I extends to a functor $\mathbf{Set} \to \mathbf{Dig}$, right adjoint to the functor V of section 1.1 which sends digraphs to their sets of vertices. Since V is a strict map of monads from **fc** to 1, taking the mate of the identity gives a natural transformation $\phi\colon \mathbf{fc} \cdot I \to I$ such that (I, ϕ) is a lax map of monads from 1 to **fc**. Then as in [13, §6.7], we can lift this adjunction first to an adjunction $V_* \dashv I_*$ between **Dig** and **fc-Gph** and then to another adjunction of the same name between **Cat** and **fc-Mult**. Explicitly, V_* sends an **fc**-graph B to the digraph of objects and vertical 1-cells of B; if B is an **fc**-multicategory, then this is a category. I_* sends a digraph D to the **fc**-graph with

- objects and vertical 1-cells given by the ojects and 1-cells of D.
- a unique horizontal 1-cell from any object to any other, and a unique horizontal 1-cell filling any rectange.

If D is a category, this is an **fc**-multicategory.

DEFINITION 3.2.2. *A* playpen *is a triple (B, C, ν), where C is a category with underlying digraph V_*B and ν is an unwiring of the map $B \xrightarrow{\eta_B} I_*C$ of* **fc***-graphs given by the unit of the adjunction $V_* \dashv I_*$ at B. A* map of playpens $(B, C, \nu) \to (B', C', \nu')$ *is a map $l\colon B \to B'$ of* **fc** *graphs such that V_*l is a functor $C \to C'$ and the diagrams*

$$\begin{array}{ccc} B \times_{I_*C} (I_*C)_0 & \xrightarrow{\nu_0} & B_0 \\ \downarrow & & \downarrow \\ B' \times_{I_*C'} (I_*C')_0 & \xrightarrow[\nu'_0]{} & B'_0 \end{array} \quad \textit{and} \quad \begin{array}{ccc} B \times_{I_*C} (I_*C \circ I_*C) & \xrightarrow{\nu_2} & B \circ B \\ \downarrow & & \downarrow \\ B' \times_{I_*C'} (I_*C' \circ I_*C') & \xrightarrow[\nu'_2]{} & B' \circ B' \end{array}$$

commute. Playpens and maps of playpens form a category **Playpen**.

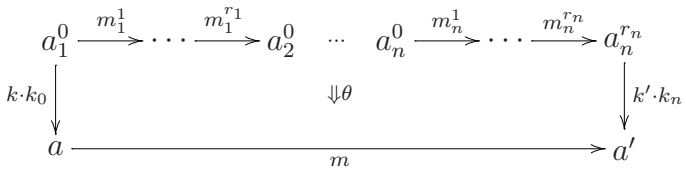
LEMMA 3.2.3. **Playpen** *has all small limits, and the forgetful functor to the category of* **fc***-graphs creates them.*

Proof. The second part of the lemma gives the construction for the first part. □

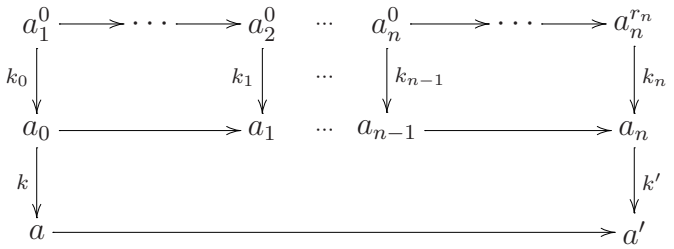
EXAMPLE 3.2.4. **fc**-graphs B with V_*B the terminal digraph 1 correspond to **fm**-graphs, where **fm** is the free monoid monad on set. Since I_*1 is the terminal **fc**-graph, this restricts to a correspondence between playpens B with V_*B the terminal category and $\mathbf{fm}^+$-arbeglas.

Following a standard convention for abusing this kind of notation, we'll often refer to playpens using the names of their underlying **fc**-graphs, when it is clear from the context what the remaining structure is. In particular, we'll refer to the underlying vertical category of a playpen B as V_*B.

Given a 2-cell

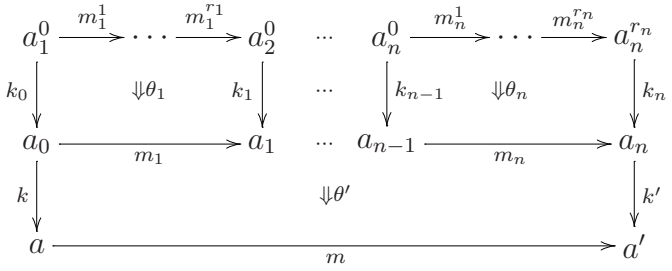


in a playpen B and a composable collection



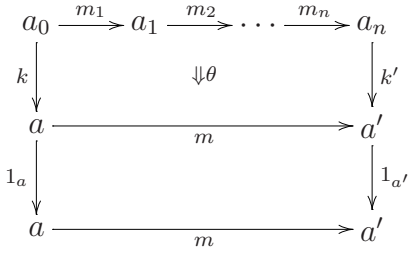
in I_*V_*B, we can decompose θ over this collection to give a composable

collection

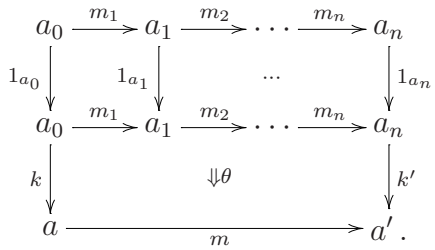


in B. Since the composable collection with respect to which θ was decomposed can be read off from this decomposition, we'll often not make explicit mention of it. Instead, we'll simply say that θ can be decomposed to give the composable collection $(\theta', (\theta_i)_{i\in[n]})$ pictured above.

By the unit laws for unwirings, any 2-cell θ must appear in two of its decompositions,



and



If θ is a unary cell whose left and right edges are identities, these decompositions coincide, and so we get the following special case of proposition 3.1.5:

LEMMA 3.2.5. *Let θ be a unary 2-cell in a playpen whose left and right edges are identities. Then the top and bottom edges of θ are equal.*

Recall the gluing construction for categories:

DEFINITION 3.2.6. *Let $l\colon A \to B$ be a functor. Then the* glued *category* $\mathbf{Gl}(l)$ *has*

- *objects given by pairs (a, f), where a is an object of A and f is a map in B with target $l(a)$.*
- *maps from (a, f) to (a', f') given by pairs (k, g), where $k\colon a \to a'$ and g are maps in A and B respectively such that $f' \cdot g = l(k) \cdot f$.*

The identities and composites are induced from those of A and B. We say that $\mathbf{Gl}(l)$ *is* obtained by glueing along l

We can, with a little effort, lift this to a gluing construction for playpens. The construction restricted to the vertical category is just normal gluing of categories. The construction on horizontal 1-cells and 2-cells could perhaps be thought of as being given by a horizontal stretching of this construction. However, since we do not have a composition of 2-cells, we cannot define the 2-cells of the glued playpen in a way strictly analogous to the definition of maps in a glued category. Instead, we must make use of the unwiring to capture a similar idea. In particular, in contrast to the case of limits, this construction is not built on a gluing construction for **fc**-graphs - the playpen structure plays an essential role.

DEFINITION 3.2.7. *Let $l\colon A \to B$ be a map of playpens. The glued playpen* $\mathbf{Gl}(l)$ *has*

- *objects given by pairs (a, f), where a is an object of A and f is a vertical 1-cell in B with target $l(a)$.*
- *vertical 1-cells from (a, f) to (a', f') given by pairs (k, g), where $k\colon a \to a'$ and g are vertical 1-cells in A and B respectively such that $f' \cdot g = l(k) \cdot f$.*

- *horizontal 1-cells from* (a, f) *to* (a', f') *given by pairs* (m, ϕ)*, where* $a \xrightarrow{m} a'$ *is a horizontal 1-cell in* A *and* ϕ *is a 2-cell in* B *filling some square*

$$\begin{array}{ccc} s(f) & \xrightarrow{s(\phi)} & s(f') \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f'} \\ l(a) & \xrightarrow[l(m)]{} & l(a')\,. \end{array}$$

- *2-cells filling the rectangle*

$$\begin{array}{ccccccc} (a_0, f_0) & \xrightarrow{(m_1,\phi_1)} & (a_1, f_1) & \xrightarrow{(m_2,\phi_2)} & \cdots & \xrightarrow{(m_n,\phi_n)} & (a_n, f_n) \\ {\scriptstyle (k,g)}\downarrow & & & & & & \downarrow{\scriptstyle (k',g')} \\ (a, f) & & \xrightarrow[(m,\phi)]{} & & & & (a', f') \end{array}$$

given by pairs (θ, ρ) *where* θ *is a 2-cell filling the rectangle*

$$\begin{array}{ccccccc} a_0 & \xrightarrow{m_1} & a_1 & \xrightarrow{m_2} & \cdots & \xrightarrow{m_n} & a_n \\ {\scriptstyle k}\downarrow & & & & & & \downarrow{\scriptstyle k'} \\ a & & \xrightarrow[m]{} & & & & a' \end{array}$$

in A*, and* ρ *is a 2-cell filling the rectangle*

$$\begin{array}{ccccccc} s(f_0) & \xrightarrow{s(\phi_1)} & s(f_1) & \xrightarrow{s(\phi_2)} & \cdots & \xrightarrow{s(\phi_n)} & s(f_n) \\ {\scriptstyle f\cdot g}\downarrow & & & & & & \downarrow{\scriptstyle f'\cdot g'} \\ l(a) & & \xrightarrow[l(m)]{} & & & & l(a') \end{array}$$

in $\mathbf{Sec}(F)$*, whose decompositions over the appropriate composable collections in* I_*V_*B *give diagrams*

$$\begin{array}{ccccccccc} s(f_0) & \xrightarrow{s(\phi_1)} & s(f_1) & \xrightarrow{s(\phi_2)} & \cdots & & \xrightarrow{s(\phi_n)} & s(f_n) \\ {\scriptstyle f_0}\downarrow & \Downarrow\phi_1 & {\scriptstyle f_1}\downarrow & \Downarrow\phi_2 & \cdots & \Downarrow\phi_n & & \downarrow{\scriptstyle f_n} \\ l(a_0) & \xrightarrow{l(m_1)} & l(a_1) & \xrightarrow{l(m_2)} & \cdots & & \xrightarrow{l(m_n)} & l(a_n) \\ {\scriptstyle l(k)}\downarrow & & & \Downarrow l(\theta) & & & & \downarrow{\scriptstyle l(k')} \\ l(a) & & & \xrightarrow[l(m)]{} & & & & l(a') \end{array}$$

and

$$\begin{array}{ccccccc}
s(f_0) & \xrightarrow{s(\phi_1)} & s(f_1) & \xrightarrow{s(\phi_2)} & \cdots & \xrightarrow{s(\phi_n)} & s(f_n) \\
{\scriptstyle g}\downarrow & & \Downarrow\widehat{\rho} & & & & \downarrow{\scriptstyle g} \\
s(f) & & \xrightarrow{\quad s(\phi)\quad} & & & & s(f') \\
{\scriptstyle f}\downarrow & & \Downarrow\phi & & & & \downarrow{\scriptstyle f'} \\
l(a) & & \xrightarrow[\quad l(m)\quad]{} & & & & l(a')\,.
\end{array}$$

The composition of vertical 1-cells is given as in $\mathbf{Gl}(V_*l)$.

Let ν^A *be the horizontal unwiring of* A*, and* ν^B *that of* B*. As in section 3.1,* ν^A *is determined by its components* ν_0^A *and* ν_2^A*, and similar comments apply to* ν^B*. We wish to specify the corresponding maps* ν_0^l *and* ν_2^l *for a horizontal unwiring* ν^l *of* $\mathbf{Gl}(l)$*.* ν^l *should be an unwiring of the map* $\eta = \eta_{\mathbf{Gl}(l)}$*, which has codomain* $I_*V_*\mathbf{Gl}(l) = I_*\mathbf{Gl}(V_*l)$.

We'll deal with ν_0^l *first. Denote the source* $\mathbf{Gl}(l)_1 \times_{(I_*\mathbf{Gl}(V_*l))_1} (I_*\mathbf{Gl}(V_*l))_0$ $= \mathbf{Gl}(l)_1 \times_{(I_*\mathbf{Gl}(V_*l))_1} IV\mathbf{Gl}(V_*l)$ *of* ν_0^l *by* P'*.* P' *is the subdigraph of* $\mathbf{Gl}(l)_1$ *involving only 1-ary 2-cells*

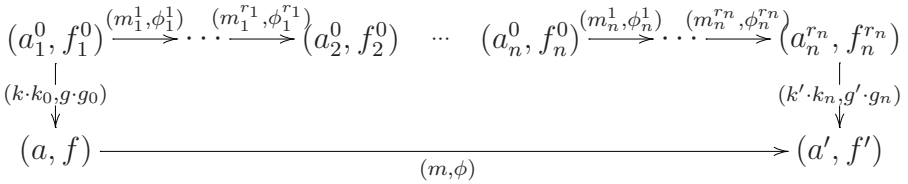
$$\begin{array}{ccc}
(a,f) & \xrightarrow{(m,\phi)} & (a',f') \\
{\scriptstyle (1_a,1_{s(f)})}\downarrow & \Downarrow(\theta,\rho) & \downarrow{\scriptstyle (1_{a'},1_{s(f')})} \\
(a,f) & \xrightarrow[(n,\psi)]{} & (a',f')
\end{array}$$

in which the right and left edges are isomorphisms. Since A *is a playpen, we must have by lemma 3.2.5 that* $m = n$*. We can decompose* ρ *a couple of different ways to get*

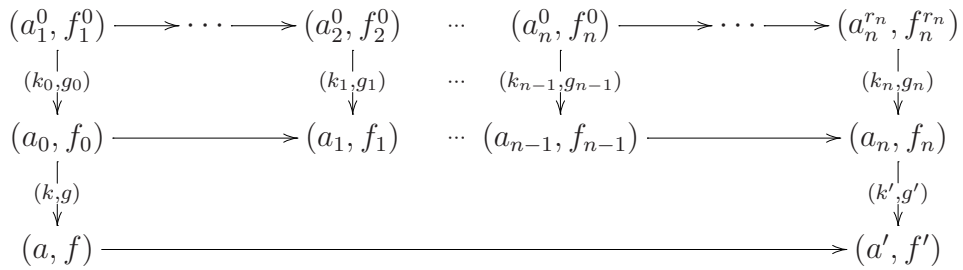
$$\begin{array}{ccc}
s(f) & \xrightarrow{s(\phi)} & s(f') \\
{\scriptstyle 1_{s(f)}}\downarrow & \Downarrow\widehat{\rho} & \downarrow{\scriptstyle 1_{s(f')}} \\
s(f) & \xrightarrow{s(\psi)} & s(f') \\
{\scriptstyle f}\downarrow & \Downarrow\psi & \downarrow{\scriptstyle f'} \\
l(a) & \xrightarrow[l(m)]{} & l(a')
\end{array}
\qquad \textit{and} \qquad
\begin{array}{ccc}
s(f) & \xrightarrow{s(\phi)} & s(f') \\
{\scriptstyle f}\downarrow & \Downarrow\phi & \downarrow{\scriptstyle f'} \\
l(a) & \xrightarrow{l(m)} & l(a') \\
{\scriptstyle 1_{l(a)}}\downarrow & \Downarrow l(\theta) & \downarrow{\scriptstyle 1_{l(a')}} \\
l(a) & \xrightarrow[l(m)]{} & l(a')
\end{array}$$

so that $\psi = \rho = \phi$*. Thus, for such a cell, we may define* $\nu(l)((\theta,\rho)) = (m,\phi) = (n,\psi)$*.*

$\nu(l)_2$ *is a little harder. Denote the source* $\mathbf{Gl}(l)_1 \times_{(I_*\mathbf{Gl}(V_*l))_1} (I_*\mathbf{Gl}(V_*l)) * (I_*\mathbf{Gl}(V_*l))$ *by* P''*. An edge of* P'' *is given by a pair* $((\theta,\rho),\underline{c})$*, where* (θ,ρ) *is a 2-cell in* $\mathbf{Gl}(l)$ *filling some rectangle*

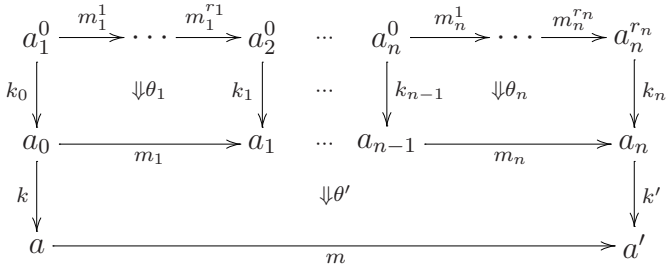


and $\underline{c}$ *is a composable collection of cells*



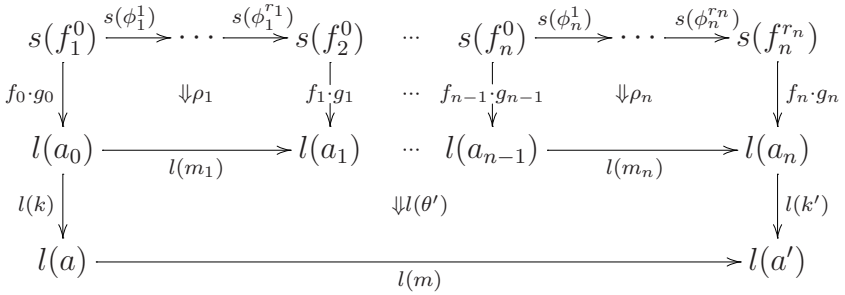
in $I_*\mathbf{Gl}(V_*l)$*.*

Since A *is a playpen, we can decompose* θ *over the part of this structure which lies in* A *to give a composable collection*

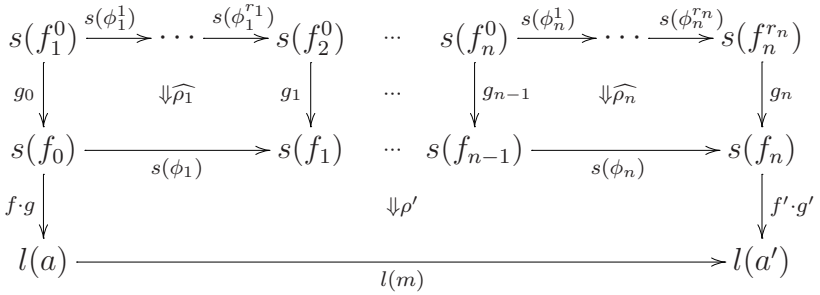


in A*.*

We can decompose ρ in a couple of different ways to get



and



where the pairs (θ_i, ρ_i) and (θ', ρ') are 2-cells in $\mathbf{Gl}(l)$.

It is clear that the composable collection that ν_2^l sends $((\theta, \rho), \underline{c})$ to should be $((\theta', \rho'), ((\theta_i, \rho_i))_{i\in[n]})$.

We say that $\mathbf{Gl}(l)$ is obtained by glueing along *the map of playpens $l\colon A \to B$.*

A playpen B lives in the slice of the category **fc-Gph** of **fc**-graphs by objects of the form I_*V_*B. It will therefore allow the lifting of exponentiation from this slice to the slice **fc-Mult**$/I^*V_*B$. There is a particular simple class of objects in these categories, arising from monoidal categories, which we will often wish to exponentiate by.

Definition 3.2.8. *Let V be a monoidal category, and let C be any category. V has an underlying plain multicategory, which we can consider as an* **fc***-multicategory $\widehat{V}$ with only one object and one map. Then there is a*

representative C_*V *of* V *in the slice* **fc-Mult**$/I_*C$*, given by* $(I_*C) \times \widehat{V}$*. More explicitly, the objects and vertical maps are those of* C*, the horizontal 1-cells from* a *to* a' *are given by the objects of* V*, and the 2-cells with top* $(m_i)_{i\in[n]}$ *and bottom* m *are given, irrespective of the maps on the left and right, by* $V(\otimes_{i\in[n]} m_i, m)$*.*

Before introducing examples of such exponentiation, we'll need to spend some time looking in detail at a couple of major examples of playpens.

3.3 The playpen MAT and the unwirable multicategory RING

There are two playpens which will play a significant role in the construction of categories of games. The first, and simpler, playpen MAT establishes a framework on which the horizontal arrows will be built. The second, more structured playpen RING controls the combinatorics of the composition of strategies.

DEFINITION 3.3.1. *The playpen* MAT *of* matrix elements *has*

- *objects given by sets.*
- *vertical 1-cells given by functions.*
- *horizontal 1-cells from* X *to* X' *given by pairs* (x, x') *with* $x \in X$ *and* $x' \in X'$*.*
- *a unique 2-cell filling the rectangle*

$$\begin{array}{ccccccc} X_0 & \xrightarrow{(x_0,x_1')} & X_1 & \xrightarrow{(x_1,x_2')} & \cdots & \xrightarrow{(x_{n-1},x_n')} & X_n \\ {\scriptstyle f}\downarrow & & & & & & \downarrow{\scriptstyle f'} \\ X & & \xrightarrow[(x,x')]{} & & & & X' \end{array}$$

if $x'_i = x_i$ for each $i \in [n-1]$, $f(x_0) = x$ and $f'(x'_n) = x'$, and no 2-cells filling this rectangle otherwise. Letting $x_n = x'_n$, we can identify this 2-cell with the sequence $(x_i)_{i\in[0,n]}$.

The composition of vertical 1-cells is the usual composition of functions. A typical decomposition of a 2-cell $(x_i)_{i\in[0,\sum_{i\in[n]} r_i]}$ is given by

$$\begin{array}{ccccccc}
X_1^0 & \longrightarrow \cdots \longrightarrow & X_2^0 & \cdots & X_n^0 & \longrightarrow \cdots \longrightarrow & X_n^{r_n} \\
{\scriptstyle f_0}\downarrow & \Downarrow (x_i)_{i\in[0,r_1]} & {\scriptstyle f_1}\downarrow & & {\scriptstyle \cdots f_{n-1}}\downarrow & \Downarrow (x_{i+\sum_{j\in[n-1]} r_j})_{i\in[0,r_n]} & \downarrow{\scriptstyle f_n} \\
X_0 & \longrightarrow & X_1 & \cdots & X_{n-1} & \longrightarrow & X_n \\
{\scriptstyle f}\downarrow & & \Downarrow (f_i(x_{\sum_{j\in[i]} r_j}))_{i\in[0,n]} & & & & \downarrow{\scriptstyle f'} \\
X & & \longrightarrow & & & & X'
\end{array}$$

Exponentiating with respect to this **fc**-graph recovers a standard construction of **fc**-multicategories. Let V be a monoidal category. Then the exponential $(\mathbf{Set}_* V)^{\mathrm{MAT}}$ has

- objects and vertical maps given by sets and functions respectively.
- horizontal 1-cells from X to Y given by indexed families $(m_{xy})_{x\in X, y\in Y}$ of objects of V.
- 2-cells filling the rectangle

$$\begin{array}{ccccccc}
X_0 & \xrightarrow{M_1} & X_1 & \xrightarrow{M_2} & \cdots & \xrightarrow{M_s} & X_s \\
{\scriptstyle f}\downarrow & & & & & & \downarrow{\scriptstyle f'} \\
X & & & & \xrightarrow[M]{} & & X'
\end{array}$$

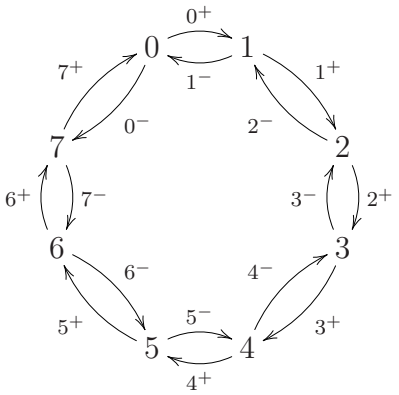
  given by families of maps $\otimes_{i\in[n]}(M_i)_{x_{i-1}x_i} \to M_{f(x_0)f'(x_n)}$ in V indexed by sequences $(x_i)_{i\in[0,n]} \in \prod_{i\in[0,n]} X_i$.
- identities and composition induced from those of V.

This is the familiar monoidal category $\mathbf{Mat}(V)$, as constructed explicitly in [4].

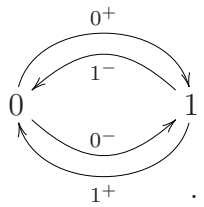
The playpen RING is rather more complicated, though its vertical structure is simpler. It only has one object and one vertical cell. Thus, as in example 3.2.4, it corresponds to an **fm**-arbegla, and it is under this aspect that we'll introduce it below. In fact, it has a little more structure than this - the unwiring is an isomorphism and so it corresponds to an unwirable plain multicategory. The structure of that plain multicategory may be given in terms of some pointed digraphs and some plays in those digraphs.

DEFINITION 3.3.2. *Let $n \in \mathbb{N}_0$. The n^{th}* ring digraph*, R_n, has vertex set $\mathbb{Z}/n\mathbb{Z}$, and for each $i \in \mathbb{Z}/n\mathbb{Z}$, R_n has an edge i^+ with source i and target $i+1$, and an edge i^- with source i and target $i-1$. These are all the edges of R_n. The basepoint is 0.*

EXAMPLE 3.3.3. R_8 is given by



and R_2 is given by



.

REMARK 3.3.4. A play in R_n can be specified by giving a sequence of +s and −s, since for any given initial sequence of moves, the next move can only be i^+ or i^-, for some i. This shows that, for example, all the trees $t(R_n)$ are isomorphic. Nevertheless, it is helpful to think of the plays in R_m and R_n as different objects.

A geometric intuition is useful for thinking about play in R_n. Imagine a loop of string with n knots on it, and with a small ring threaded onto it. Now imagine a simple game in which the ring is moved around on the string, with a single move being to pull the ring past a single knot. This game corresponds to R_n, with the positions corresponding to the sections of string between successive knots. Now if you look at play in the string game corresponding to R_n, but ignoring k of the knots, then what you see looks like play in the string game R_{n-k}. This can be made precise as follows:

DEFINITION 3.3.5. *For any natural number n, the set* gap_n *of* gaps *in* $\mathbb{Z}/n\mathbb{Z}$ *is given by* $\left(\left(\frac{1}{2}\mathbb{Z}\right)/n\left(\frac{1}{2}\mathbb{Z}\right)\right) \setminus (\mathbb{Z}/n\mathbb{Z})$. *The element* $i + \frac{1}{2}$ *is called the* gap between i *and* $i+1$. *For any edge e of R_n there is a* corresponding gap $g(e)$, *given by* $g(i^\pm) = i \pm \frac{1}{2}$. *The move e is* in *the gap $g(e)$.*

DEFINITION 3.3.6. *Let f be an injective map from* $\mathrm{gap}_{n'}$ *to* gap_n *which preserves the cyclic order. Then there is an injection* $f_E \colon E(R_{n'}) \to E(R_n)$, *sending* $i^\pm$ *to* $\left(f(g(i)) \mp \frac{1}{2}\right)^\pm$, *so that* $g \cdot f_E = f \cdot g$. *For any play p in R_n, the* restriction $p\restriction_f$ *of p along f is the play in $R_{n'}$ given by applying f_E^{-1} termwise to p, discarding any terms not in the image of f_E*

This could be phrased another way. For each $i \in \mathbb{Z}/n\mathbb{Z}$, there is a unique $i' \in \mathbb{Z}/n'\mathbb{Z}$ such that i lies between $f(i' - \frac{1}{2})$ and $f(i' + \frac{1}{2})$ in the cyclic order. This gives a map $f^* \colon \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n'\mathbb{Z}$, which extends to a map $R_n \to \mathbf{fc}(R_{n'})$ by

$$e \mapsto \begin{cases} e' & \text{when } \exists e' \colon f_E(e') = e \\ 1_{f^*(s(e))} & \text{otherwise} \end{cases}.$$

This in turn extends to a functor $\mathbf{fc}(R_n) \to \mathbf{fc}(R_{n'})$ and so induces the map $-\restriction_f \colon \mathbf{Play}(R_n) \to \mathbf{Play}(R_{n'})$.

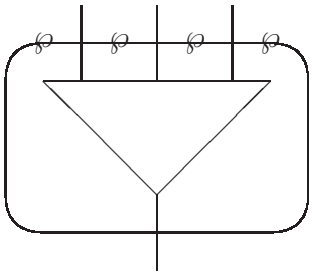
This can easily be generalised to the case where f is not an injection.

DEFINITION 3.3.7. *Let $f \colon \mathrm{gap}_{n'} \to \mathrm{gap}_n$ preserve the cyclic order. For each $e \in E(R_n)$, the set of edges in $f_E^{-1}(e)$ form a composable sequence; sending each such edge e to the corresponding sequence gives a basepoint-preserving map $R_n \to \mathbf{fc}(R_{n'})$. This induces a map $-\restriction_f \colon \mathbf{fc}(R_n) \to \mathbf{fc}(R_{n'})$, called* restriction *along f.*

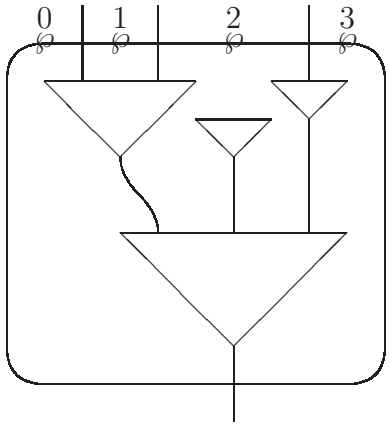
We shall mostly be interested in the restriction of this map to domain $\mathbf{Play}(R_n)$ and codomain $\mathbf{Play}(R_{n'})$.

How does this generalisation appear from the intuitive perspective in terms of knots on string? Given loops L and L' of string with n and n' knots on them respectively (these knots correspond to the gaps), and a map f as in definition 3.3.7, f extends from a map of knots to a map of strings from L' to L. For any two consecutive knots which are sent to the same knot, the section of string joining them is squashed down to an infinitesimal size. Thus for any movement of the ring past a knot k on L', the corresponding movement of the ring on L takes it past all the knots mapping to k (and the sections joining them) in a single sweep.

There are some gap maps f which will be especially useful in the construction of of an unwirable multicategory below. The key to visualising the relationship between string games and plain multicategories is to imagine draping loops of string around the outside of the cells of plain multicategories, so that the knots separate out the inputs and outputs, which each lie in their own gap, like this:



Now, if this is done with a composable collection of cells, for example



then to each n-ary cell we can associate an $n+1$-tuple of knots on the outer string, namely those in the same connected components of the plane as the sections of boundary of that cell. For example, in the picture above the cells in the top row get the tuples 012, 2 and 23 respectively, and the cell in the bottom row gets the tuple 0223. So for each such composable collection, and each cell in such a collection, there is a corresponding gap map.

Definition 3.3.8. *Let* 1 *be the terminal multicategory, and* $s = (n, (n_i)_{i\in[n]})$ *be a composable collection of cells in* 1*, as in example 3.1.10. The* thread maps *are given by*

$$\mathrm{Thr}_s \colon \mathrm{gap}_{n+1} \to \mathrm{gap}_{\sum_{j\in[n]} n_j+1} \qquad i + \tfrac{1}{2} \mapsto \sum_{j\in[i]} n_j + \tfrac{1}{2}$$

$$\mathrm{Thr}_{s,j} \colon \mathrm{gap}_{n_j+1} \to \mathrm{gap}_{\sum_{j\in[n]} n_j+1} \qquad g \mapsto g + \sum_{i\in[j-1]} n_i$$

and for any natural number n by

$$\mathrm{Thr}_n = \mathrm{Thr}_{(1,(n))}$$

$$\mathrm{Thr}_{n,j} = \mathrm{Thr}_{(n,(1)_{i\in[n]}),j}$$

So, for example, $\mathrm{Thr}_{n,j}$ is the map from gap_2 to gap_{n+1} sending $\frac{1}{2}$ to $j-\frac{1}{2}$ and $\frac{3}{2}$ to $j+\frac{1}{2}$.

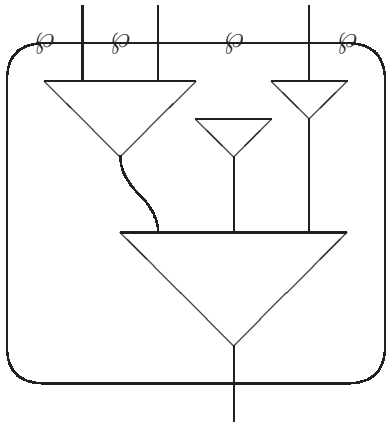
We are now in a position to define the underlying $\mathbf{fm}^+$-graph of RING.

DEFINITION 3.3.9. *The $\mathbf{fm}^+$-graph* RING *has*

- *objects given by plays in R_2.*
- *maps with source $(p_i)_{i\in[n]}$ and target p given by plays ϕ in R_{n+1} such that $\phi\restriction_{\mathrm{Thr}_{n,i}} = p_i$ and $\phi\restriction_{\mathrm{Thr}_n} = p$.*

It isn't too hard to see what the unwiring ν of RING should be: First, we need a map $\nu_0 \colon \mathrm{RING}_1 \times 1_0 \to \mathrm{RING}_0$. Here 1_0 is the $\mathbf{fm}^+$-graph with a single object and a single map of arity 1. Thus $\mathrm{RING}_1 \times 1_0$ is the sub-$\mathbf{fm}^+$-graph of RING consisting of those maps of arity 1. Such a map is a play p in R_2; the source and target are both also equal to p. So sending each such play p to itself gives such a map ν_0, and this map is evidently an isomorphism. $\nu_2 \colon \mathrm{RING} \times (1 \circ 1) \to \mathrm{RING} \circ \mathrm{RING}$ is only a little harder. Using the notation of definition 3.1.9, the decomposition $(c_s, (c_{s,i})_{i\in[n]})$ of the cell c over the shape s should be given by $c_s = c\restriction_{\mathrm{Thr}_s}$ and $c_{s,i} = c\restriction_{\mathrm{Thr}_{s,i}}$. In fact, this map is defined for all plays c, not just those with basepoint 0.

The structure so far gives an unwiring of RING, but there is more to be said. Imagine some play in a string game, with the string draped around a composable collection of cell-shapes as in the picture



considered earlier. The knots in the tuple associated to the bottom cell divide the top of the string into segments, which we can think of as being controlled by the cells in the top row. If we know the restriction of the play to the knots in this tuple, then we know how the ring moves between these segments. If we also know the restriction of the play to the knots in the tuple associated to a particular cell in the top row, then we know how the ring moves when in the segment controlled by that cell. So given the restrictions of the play to the knots in each of these tuples, we can reconstruct the play itself. But this information is precisely that given in the decomposition of the play over this composable collection. Further, given any compatible information of this kind (how to move the ring between the segments controlled by the cells in the top row, and how to move it within each such segment), we can paste it together to find a play in the ring game which decomposes to give that information. It follows that the unwiring specified above is an isomorphism. We may make this argument a little more precise.

DEFINITION 3.3.10. *For a cell c of* RING*, the* length $l(c)$ *of c is the number of moves in c. Let $\underline{c} = (c', (c_i)_{i\in[n]})$ be a composable collection of cells in* RING*. The* shape $\sigma(\underline{c})$ *of $\underline{c}$ is $!*!(\underline{c})$ The* length $\lambda(\underline{c})$ *of $\underline{c}$ is $\sum_{j\in[n]}(l(c_j) - l(c_j\restriction_{\mathrm{Thr}_{n_j+1}})) + l(c')$.*

LEMMA 3.3.11. *Let* (c, s) *be a cell in* RING$\times\mathbf{fm}^+1$. *Then* $\sigma((c_s, (c_{s,i})_{i\in[n]})) = s$ *and* $\lambda((c_s, (c_{s,i})_{i\in[n]})) = l(c)$.

Proof. The first equation follows from the fact that ν is an unwiring. For the second, note that the number of moves in c in the gaps from $\sum_{i\in[j-1]} n_i + \frac{3}{2}$ to $\sum_{i\in[j]} n_i - \frac{1}{2}$ is given by the number of moves in the gaps from $\frac{3}{2}$ to $n_j - \frac{1}{2}$ in $c_{s,j}$, that is, by $l(c_{s,j}) - l(c_{s,j}\restriction_{\mathrm{Thr}_{n_j+1}})$. The moves not counted so far are those in the gaps $\sum_{i\in[j]} n_i + \frac{1}{2}$, and there are $l(c_s)$ of these. □

THEOREM 3.3.12. ν_2 *is an isomorphism*

Proof. It is enough to show (for a fixed shape s) that for each $k \in \mathbb{N}_0$ the restriction of ν_2 to the set of pairs $(c, s) \in$ RING $\times\ \mathbf{fm}^+1$ with $l(c) = k$ is a bijection onto the collection of composable collections $\underline{c}$ of shape s and length k. We shall prove this by induction.

For the base case ($k = 0$), observe that if $\underline{c}$ is a composable collection of length 0 then in particular $l(c')$ must be 0 and so c' must be the empty play. But then each $l(c_j\restriction_{\mathrm{Thr}_{n_j+1}})$ is also 0, and so each $l(c_j)$ is also 0; that is, all the plays comprising $\underline{c}$ must have length 0. But there is also a unique play c of length 0.

For the induction step, let $\underline{c}$ be a composable collection of shape s and length k. It is necessary to reconstruct some c with $\nu_2(c, s) = \underline{c}$. Let $j = t(c')$, and suppose for the present that $j \neq 0$. The target of any suitable c must lie in the range $(\sum_{i\in[j-1]} n_i, \sum_{i\in[j]} n_i]$ and so the final move of c must be $m = m_j + \sum_{i\in[j-1]} n_i$, where m_j is the last move of c_j. Each term of $\nu_2((m), s)$ is either a single move or an empty sequence, and it is simple to check that in each case where $(m)_{s,j}$ (resp. $(m)_s$) is a move it is the final move of c_j (resp. c), and that removing these final moves gives a composable collection $\widehat{\underline{c}}$ of length $k-1$ and shape s. By the induction hypothesis, there is a unique $\widehat{c}$ with $\nu_2(\widehat{c}, s) = \widehat{\underline{c}}$. c can only be the play obtained by adjoining m to the end of $\hat{c}$, but this works since ν_2 respects composition of plays.

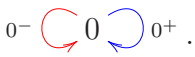
The case $j = 0$ is similar and, if anything, easier. The final move must be 1^- or $(-1)^+$ according as the final move of c' is 1^- or $(-1)^+$, and we may proceed as before. □

In particular, it follows that RING has the structure of a plain multicategory. The composite of a composable collection of cells is the unique cell which decomposes into that collection.

3.4 Application to digraphs

Combining some of these constructions, we can now build a fairly normal-looking **fc**-multicategory of games. However, it will take a little careful unwrapping to see why this is what the constructions produce.

The pointed digraph R_1 has one object and two maps as in the picture



The two edges have been coloured red and blue to indicate their correspondence with the two players, traditionally called red and blue, who we will consider to be playing the games. The way in which the roles of the players correspond to these edges will only emerge slowly.

For any play p in the n^{th} string game, we can examine how it looks from the perspective of each knot. Ignoring all knots but the i^{th} gives a play in R_1 for each i. This gives an n-tuple of plays in R_1 for each play in R_n. Putting all this together gives a map of playpens from RING to MAT.

DEFINITION 3.4.1. *The i^{th}* knot map $\text{Knot}_i^n \colon \text{gap}_0 \to \text{gap}_n$ *sends* $\frac{1}{2}$ *to* $i + \frac{1}{2}$.

DEFINITION 3.4.2. *The map* KNOT : RING $\to$ MAT *sends*

- *the unique object to* $\mathbf{Play}(R_1)$
- *the unique vertical map to the identity on* $\mathbf{Play}(R_1)$

- *the horizontal 1-cell p to* $(p\restriction_{\mathrm{Knot}_0^2}, p\restriction_{\mathrm{Knot}_1^2})$
- *the n-ary 2-cell c to* $(c\restriction_{\mathrm{Knot}_i^{n+1}})_{i\in[0,n]}$.

Corresponding to the object R_1 of $\mathbf{Dig}_*$, there is a functor $R_1 \colon V_*\textsc{ring} = 1 \to \mathbf{Dig}_*$, and transposing across the adjunction $V_* \dashv I_*$ we get a map $\overline{R_1}$ such that the square

$$\begin{array}{ccc} \textsc{ring} & \xrightarrow{\textsc{knot}} & \textsc{mat} \\ {\scriptstyle \overline{R_1}}\downarrow & & \downarrow{\scriptstyle \eta} \\ I_*\mathbf{Dig}_* & \xrightarrow[I_*\mathbf{Play}]{} & I_*\mathbf{Set} \end{array}$$

commutes. This in turn gives a map $\textsc{alt} \colon \textsc{ring} \to P$, where P is the pullback of the lower and right edges of this square. It will be an exponential of $\mathbf{Gl}(\textsc{alt})$ which will be our first serious example of a multicategory of games. However, before taking that exponential, let's pause to unpeel the structure of P and of $\mathbf{Gl}(\textsc{alt})$.

P is easy enough to understand. The category of objects and vertical maps is $\mathbf{Dig}_* \times_{\mathbf{Set}} \mathbf{Set} = \mathbf{Dig}_*$. A horizontal 1-cell from D to D' consists of a pair (p, p') with p a play in D and p' a play in D'. 2-cells are given by tuples of plays, as in

$$\begin{array}{ccccccc} D_0 & \xrightarrow{(p_0,p_1)} & D_1 & \xrightarrow{(p_1,p_2)} & \cdots & \xrightarrow{(p_{n-1},p_n)} & D_n \\ {\scriptstyle f}\downarrow & & \Downarrow (p_i)_{i\in[0,n]} & & & & \downarrow{\scriptstyle f'} \\ D & & \xrightarrow[(f(p_0),f'(p_n))]{} & & & & D' \end{array}$$

with decompositions as in $\textsc{mat}$.

$\textsc{alt}$ sends the unique object of $\textsc{ring}$ to R_1, and the unique vertical 1-cell to the identity on R_1. The action on horizontal 1-cells and on 2-cells is as in definition 3.4.2.

The objects of $\mathbf{Gl}(\textsc{alt})$ are given by pairs (a, f), where a is the unique object of $\textsc{ring}$, and f is a map of pointed digraphs with target $\textsc{alt}(a) = R_1$. Since a is fixed, we'll normally refer to the object as f in this section. f must

send each object of the pointed digraph $s(f)$ to $*$, and it sorts the edges into two classes; the *red* edges, which it maps to 0^-, and the *blue* edges, which it maps to 0^+. So the objects of $\mathbf{Gl}(\textsc{alt})$ correspond to pointed digraphs in which each edge has been coloured either red or blue; we'll call these *bicoloured digraphs*. In future sections, we'll sometimes abuse notation by referring to a bicoloured digraph by the name of its underlying digraph.

If pointed digraphs can be thought of as rudimentary games, then bicoloured digraphs can be thought of as slightly more normal games, to be played between the two players red and blue. The red edges are thought of as possible moves for red, and the blue edges as possible moves for blue.

The vertical 1-cells from f to f' are given by pairs (k, g), where k is the unique vertical 1-cell in RING and so can be neglected; we'll normally refer to such a 1-cell as g. $g\colon s(f) \to s(f')$ is a map of pointed digraphs such that $f' \cdot g = l(k) \cdot f = f$. That is, the category of objects and vertical 1-cells in $\mathbf{Gl}(\textsc{alt})$ is isomorphic to the slice category $\mathbf{Dig}_*/R_1$, which we'll call the category of bicoloured digraphs. A map in this category is a map of pointed digraphs sending red edges to red edges and blue edges to blue edges.

The horizontal 1-cells from $D \xrightarrow{f} R_1$ to $D' \xrightarrow{f'} R_1$ in $\mathbf{Gl}(\textsc{alt})$ are given by pairs (p, ϕ), where p is a horizontal 1-cell in RING and ϕ is a 2-cell in P, filling some square

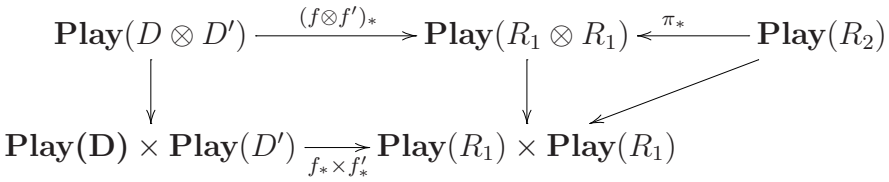
$$\begin{array}{ccc} D & \xrightarrow{(p_0,p_1)} & D' \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f'} \\ R_1 & \xrightarrow[\textsc{alt}(p)]{} & R_1\,. \end{array}$$

These correspond to a more familiar kind of object, though the way in which they do so takes a little bit of unwrapping.

First of all, we must have $\phi = (p_0, p_1)$, and so $\textsc{alt}(p) = (f_* p_0, f'_* p_1)$. p can be recovered from this, together with the information about the order in which the moves contributing to $(f_* p_0, f'_* p_1)$ are interleaved. But this interleaving information also gives a way to interleave the plays p_0 and p_1 to

give a play in $s(f) \otimes s(f')$.

More precisely, we have a diagram



in which the square on the left is a pullback, and π is the map sending moves past the i^{th} knot to moves in the i^{th} component and preserving the direction (clockwise or anticlockwise) of moves. A horizontal 1-cell from f to f' consists of an element of $\mathbf{Play(D)} \times \mathbf{Play}(D')$ and one of $\mathbf{Play}(R_2)$ mapping to the same thing in $\mathbf{Play}(R_1) \times \mathbf{Play}(R_1)$, or equivalently to an element σ of $\mathbf{Play}(D \otimes D')$ whose image under $(f \otimes f')_*$ is in the image of the injective map π_*. Since σ completely determines the 1-cell, we shall subsequently refer to the 1-cell itself as σ.

Now let Q be the pointed digraph $0 \rightleftarrows 1$, with basepoint 0. The colouring $0 \rightleftarrows 1$ specifies a map $o\colon Q \to R_1$. A play in R_1 is of the form o_*p iff all odd numbered edges are red and all even numbered edges are blue. We'll call such plays in R_1 *orderly*; they follow the standard convention of alternating play with red making the first move. This idea can be extended to bicoloured digraphs in general. Let $f\colon D \to R_1$ be a bicoloured digraph. We'll call a play ϕ in D *orderly* if $f_*\phi$ is orderly. Once more, this captures the convention of alternating play with red making the first move.

Next, let $\Rightarrow$ be the map $R_1 \otimes R_1 \to R_1$ with the following action on edges:

$$\begin{aligned} (0^+, 0) &\mapsto 0^- \\ (0^-, 0) &\mapsto 0^+ \\ (0, 0^+) &\mapsto 0^+ \\ (0, 0^-) &\mapsto 0^- \end{aligned}$$

so that edges in the first component are mapped to the edge of opposite colour, whereas edges in the second component are mapped to the edge of the same colour. This map encodes another standard convention in the construction of categories of games; a map from G to H is normally taken to be a strategy in the game $G \Rightarrow H$ in which G and H are played simultaneously, but with the roles of the players exchanged for play in G. If $D \xrightarrow{f} R_1$ and $D' \xrightarrow{f'} R_1$ are bicoloured digraphs, we can define this combination $f \Rightarrow f'$ by $D \otimes D' \xrightarrow{f \otimes f'} R_1 \otimes R_1 \xrightarrow{\Rightarrow} R_1$.

What does all of this have to do with the horizontal 1-cells of $\mathbf{Gl}(\textsc{alt})$? Well, it is elementary to check that there is a pullback square

$$\begin{array}{ccc} R_2 & \xrightarrow{\pi_2} & R_1 \otimes R_1 \\ \downarrow & & \downarrow \Rightarrow \\ Q & \xrightarrow[o]{} & R_1 \,. \end{array}$$

Now suppose we have a horizontal 1-cell σ from $D \xrightarrow{f} R_1$ to $D' \xrightarrow{f'} R_1$ in $\mathbf{Gl}(\textsc{alt})$. Recall that this is a play in $D \otimes D'$ whose image under $(f \otimes f')_*$ is in the image of π_*. Considering the pullback above, this is equivalent to the condition that there is some play p' in Q such that $o_* p' = (\Rightarrow)_*(f \otimes f')_* \sigma = (f \Rightarrow f')_* \sigma$. That is, the condition is that σ, considered as a play in $D \Rightarrow D'$, is orderly. So the conclusion is that horizontal maps from D to D' correspond to orderly plays σ in $D \Rightarrow D'$.

2-cells filling the rectangle

$$\begin{array}{ccccccc} (a_0, f_0) & \xrightarrow{(p_1,\phi_1)} & (a_1, f_1) & \xrightarrow{(p_2,\phi_2)} & \cdots & \xrightarrow{(p_n,\phi_n)} & (a_n, f_n) \\ {\scriptstyle (k,g)}\downarrow & & & & & & \downarrow{\scriptstyle (k',g')} \\ (a, f) & & \xrightarrow[(p,\phi)]{} & & & & (a', f') \end{array}$$

in $\mathbf{Gl}(\textsc{alt})$ are given by pairs (θ, ρ), where θ is a cell in RING with source

$(p_i)_{i\in[n]}$ and target p, and ρ is a 2-cell filling the rectangle

$$\begin{array}{ccccccc} s(f_0) & \xrightarrow{s(\phi_1)} & s(f_1) & \xrightarrow{s(\phi_2)} & \cdots & \xrightarrow{s(\phi_n)} & s(f_n) \\ {\scriptstyle g}\downarrow & & & & & & \downarrow{\scriptstyle g'} \\ s(f) & & & & \xrightarrow[s(\phi)]{} & & s(f') \end{array}$$

in P satisfying certain conditions.

As with horizontal 1-cells, such 2-cells are given by plays Φ in $\bigotimes_{i\in[0,n]} s(f_i)$ satisfying some conditions. The first condition is that there is some play θ in R_{n+1} such that $(\bigotimes_{i\in[0,n]} f_i)_*\Phi = \pi_*\theta$, where as above $\pi\colon R_{n+1} \to R_1^{\otimes(n+1)}$ is the map sending moves past the i^{th} knot to moves in the i^{th} component and preserving the direction (clockwise or anticlockwise) of moves. We'll call plays satisfying this condition *simulations*. There are a few alternative characterisations of simulations, but we won't go into the details here. It is worth noting that one putative condition, namely that, for each i, the list of moves in the $i-1^{\text{st}}$ and i^{th} components, considered as a play in $s(f_{i-1}) \otimes s(f_i)$, should be orderly, is too weak to characterise simulations. In the intuitive conception of the construction of categories of games outlined in the introduction, composition is defined by consideration of play in 3 games at once. Simulations give a formalisation of the similar intuitive conception of play in many games at once.

To state the remaining conditions concisely, a notation for considering the parts of plays in tensor products of digraphs which lie in each component will be helpful.

Definition 3.4.3. *If p is a play in $\bigotimes_{i\in I} D_i$ and $J \subseteq I$, then $x\restriction_J$ is the play in $\bigotimes_{i\in J} D_i$ consisting of those moves of p which are moves in some component D_i with $i \in J$, taken in the same order in which they appear in p.*

If $n > 0$, the remaining conditions state that, for each $i \in [n]$, $\Phi\restriction_{\{i-1,i\}}$ must be the play σ_i corresponding to (p_i, ϕ_i) and that $(f \otimes f')_*(\rho\restriction_{\{0,n\}})$ must

be the play σ corresponding to (p, ϕ). We'll say that a simulation Φ satisfying the first of these conditions at i *follows* σ_i, and that a simulation Φ satisfying the second of these conditions *follows* σ.

The conditions for nullary 2-cells are slightly different. The first of the conditions mentioned in the last paragraph becomes vacuous, but the second becomes nonsense. It should be replaced by the condition that $\sigma\restriction_{\{0\}} = \sigma\restriction_{\{1\}} = \Phi$, and a restriction on σ itself. Recall that Thr_0 is the unique map from gap_2 to gap_1. Then the condition is that $(f \otimes f')_*\sigma$ should be of the form $\pi_*(p\restriction_{\mathrm{Thr}_0})$ for some play p in R_1. If p has length t then $\pi_*(p\restriction_{\mathrm{Thr}_0})$ has length $2t$, and for each $i \in [t]$ the $2i - 1^{\mathrm{st}}$ and $2i^{\mathrm{th}}$ moves are in opposite components. All this implies that σ must have the same even length $2t$, and that for each $i \in [t]$ the $2i^{\mathrm{th}}$ move in σ is the same move as the $2i - 1^{\mathrm{st}}$, but made in the opposite component. That is σ must be one of the plays which appear in the construction of identity strategies in the standard intuitive construction of categories of games. We'll call such plays *copycat plays*.

The restriction maps of definition 3.4.3 arise from a colax monoidal structure on the functor **Play** - though this structure is not used in the construction of $\mathbf{Gl}(\textsc{alt})$, it is helpful for understanding it. In fact, this structure may be used to give a different characterisation of ALT from which the results of this section follow a little more easily, and a similar construction works in the context of a cartesian colax monoidal functor to **Set** (in fact, even in a slightly more general context). This construction is discussed in appendix A.

We are now ready to examine the construction of the **fc**-multicategory of digraph games. Since $\mathbf{Gl}(\textsc{alt})$ is a playpen, we may (as in section 2.3) use the unwiring to lift the operation of exponentiating with respect to $\mathbf{Gl}(\textsc{alt})$ from the category of **fc**-graphs over $I_*\mathbf{Dig}_*/R_1$ to the category of **fc**-multicategories over the same thing. Taking a very simple case of this, let **2** be the monoidal category corresponding to the 2-point lattice, and let $(\mathbf{Dig}_*)_*\mathbf{2}$ be its representative in this slice, as in definition 3.2.8. The

exponential $((\mathbf{Dig}_*)_*\mathbf{2})^{\mathbf{Gl}(\mathrm{ALT})}$ is an **fc**-multicategory **DigGam** with

- objects and vertical 1-cells given by the objects and maps of the category $\mathbf{Dig}_*/R_1$ of bicoloured digraphs.
- horizontal 1-cells from D to D' given by sets σ of orderly plays in $D \Rightarrow D'$. These can be thought of as rudimentary strategies; the set σ is thought of as the set of plays which might arise when playing according to the strategy. Remarkably, even these rudimentary strategies will turn out to have a well-defined composition.
- a unique 2-cell filling the rectangle

$$\begin{array}{ccccccc} D_0 & \xrightarrow{\sigma_1} & D_1 & \xrightarrow{\sigma_2} & \cdots & \xrightarrow{\sigma_n} & D_n \\ {\scriptstyle f}\downarrow & & & & & & \downarrow{\scriptstyle f'} \\ D & & & & \xrightarrow[\sigma]{} & & D' \end{array}$$

  when each simulation following elements of the respective σ_i must also follow an element of σ, and no 2-cell filling this rectangle otherwise.
- a unique nullary 2-cell filling the rectangle

$$\begin{array}{ccc} D_0 & = & D_0' \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f} \\ D & \xrightarrow[\sigma]{} & D \end{array}$$

  when each copycat play in $D_0 \Rightarrow D_0$ follows an element of σ.

The horizontal structure already looks remarkably like a standard category of games. However, no composition has been defined on this digraph yet. This composition arises from the universal properties of some cells of **DigGam**, which relies on the fact that **DigGam** is *representable*. In the next chapter, therefore, we shall discuss representability for **fc**-multicategories.

Chapter 4

Representability

There are many cases where multicategories produced by exponentiation have additional good properties. For example, exponentiating **Set** by a discrete promonoidal category gives not just a plain multicategory, but a monoidal category. Every monoidal category has an underlying plain multicategory, and in fact monoidal categories correspond to plain multicategories in which there are cells with certain universal properties. In such cases, we say the underlying plain multicategory is *representable* The details of this approach to monoidal categories can be found in [13, §3.3].

This fact is part of a more general pattern, which is outlined in [4]: often there is a forgetful bifunctor from pseudo T-algebras for some pseudomonad T into a suitable bicategory of multicategories. The multicategories corresponding to pseudo T-algebras are those with particular lifting properties. Once more, they are referred to as *representable* multicategories (of the appropriate kind).

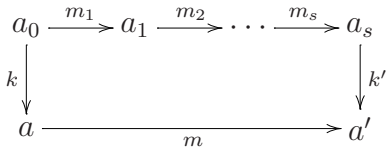
Representable **fc**-multicategories correspond in this way to weak double categories. Once more, **fc**-multicategories produced by exponentiation can often turn out to be representable. This doesn't always happen; for example the **fc**-multicategories $\mathbf{Mat}(V)$ constructed just after definition 3.3.1 may not

be representable if V doesn't have coequalisers preserved on both sides by $\otimes$. In this chapter, we'll examine a very special case in which **fc**-multicategories produced by exponentiation turn out to be representable. We'll only consider exponentials of **2** by the kinds of **fc**-multicategories produced by the gluing construction of the last chapter. In order to show that these are representable, we'll need to make use of a slightly technical property of the **fc**-multicategories produced by gluing.

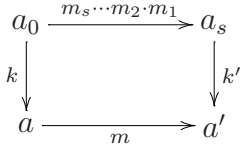
In section 4.1, we'll explain the details of representability for **fc**-multicategories. Then in section 4.2, we'll use these details to motivate the slightly technical property we need. In section 4.3, we'll demonstrate that exponentials of this special kind are representable, and explore how this works in a couple of cases. Finally, in section 4.4, we'll explain a link to slice constructions for **fc**-multicategories.

4.1 Opcartesian and weakly opcartesian 2-cells

Any weak double category C has an *underlying* **fc**-multicategory, with the same objects and vertical and horizontal 1-cells as C, and with the 2-cells filling the rectangle

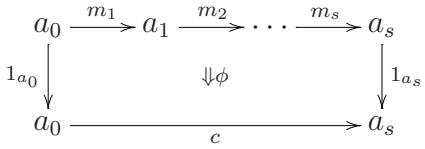


given by 2-cells filling the square

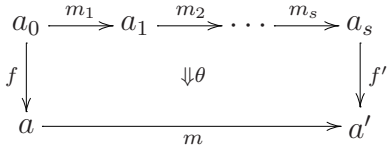


in C, with identities and composites induced from those of C. The **fc**-multicategories which arise (up to isomorphism) in this way are called *representable* **fc**-multicategories, and can be characterised by the presence of cells with certain universal properties. One presentation of this result is as follows:

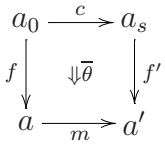
A 2-cell



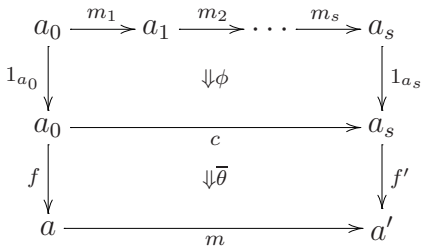
in an **fc**-multicategory C is *weakly opcartesian* iff for any other 2-cell



there is a unique 2-cell

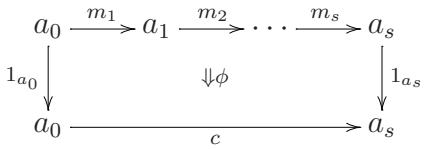


such that the composite

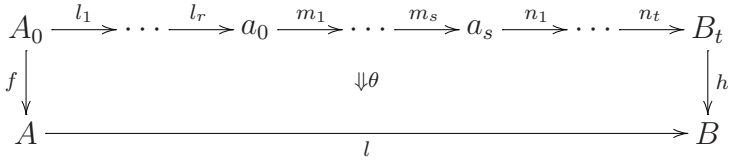


is equal to θ. In such a case, c is called a *pre-composite* of the sequence $(m_i)_{i\in[0,s]}$. If $s = 0$, we also call c a *pre-unit* at a_0. Evidently, pre-composites are unique up to unique isomorphism. If every such sequence in C has a pre-composite, then we say C *has pre-composites*.

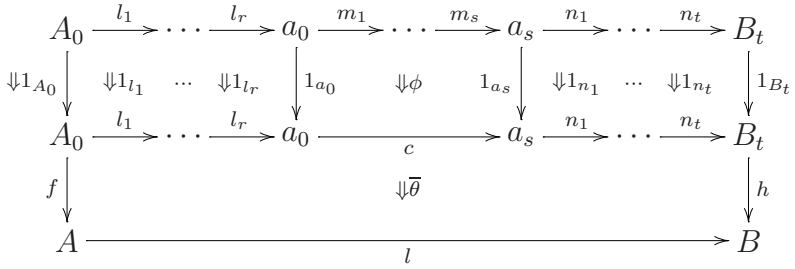
A 2-cell



in an **fc**-multicategory C is *opcartesian* iff for any pair of sequences given by $A_0 \xrightarrow{l_1} A_1 \xrightarrow{l_2} \cdots \xrightarrow{l_r} A_r$ and $B_0 \xrightarrow{n_1} B_1 \xrightarrow{n_2} \cdots \xrightarrow{n_t} B_t$ such that $A_r = a_0$ and $a_s = B_0$ and any 2-cell



there is a unique 2-cell $\overline{\theta}$ such that the composite



is equal to θ. In such a case, c is called a *composite* of the sequence $(m_i)_{i\in[0,s]}$. If the sequence has length 0 (with source and target X), c is called a *unit* at X. Evidently, opcartesian 2-cells are weakly opcartesian, and so composites and units are unique up to unique isomorphism.

An **fc**-multicategory C is the underlying **fc**-multicategory of some weak double category iff every such sequence $(m_i)_{i\in[0,s]}$ has a composite, or equivalently iff every such sequence has a pre-composite and any composite of weakly opcartesian cells is again weakly opcartesian. In such cases, C is called a *representable* **fc**-multicategory. This situation, with two equivalent conditions, one of which involves a strong universal property and the other of which involves a weaker universal property but specifies that cells with

this property should be closed under composition, is reminiscent of the pair of equivalent definitions of a fibration for a category. Indeed, this kind of phenomenon often occurs in the context of representability, mentioned at the beginning of this chapter, which generalises both of these cases.

The composition induced as in this section is usually only weak, but in the cases we'll consider later in this chapter, the representable multicateogories have a property which forces these composites to be associative and unital 'on the nose'. An **fc**-multicategory C is *locally ordered* iff there is at most one 2-cell filling any given rectangle in C. For each pair (A, A') of objects of an **fc**-multicategory C, there is a category $C(A, A')$ with objects given by horizontal 1-cells from A to A' and maps given by 2-cells between them. If C is locally ordered, then each category $C(A, A')$ becomes a partial order (hence the name). Each arch

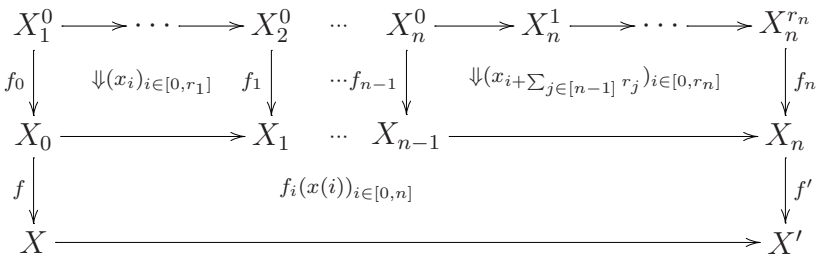
$$\begin{array}{ccccccc} a_0 & \xrightarrow{m_1} & a_1 & \xrightarrow{m_2} & \cdots & \xrightarrow{m_s} & a_s \\ {\scriptstyle 1_{a_0}}\downarrow & & & & & & \downarrow{\scriptstyle 1_{a_s}} \\ a_0 & & & & & & a_s \end{array}$$

determines an upwards-closed subset of $C(a_0, a_s)$, given by those 1-cells m for which there is a 2-cell filling the rectangle given by adding m to the bottom of the arch. There is a weakly opcartesian cell filling the arch iff this upset has a least element.

There is no guarantee that these weakly opcartesian cells will be closed under composition, but if they are then C is a locally ordered weak double category (that is, a weak double category with at most one 2-cell filling any square). But since all the structural isomorphisms must be identities (by uniqueness), any locally ordered weak double category is a strict double category. In this way, any locally ordered representable **fc**-multicategory corresponds to a strict double category. It is from the horizontal composition in such contexts that the composition of strategies in categories of games arises.

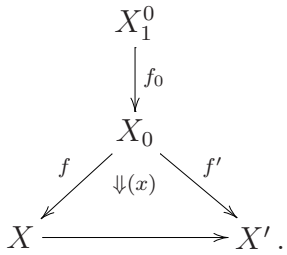
4.2 Composition in playpens

We must now make a brief digression to consider how close the playpens considered in the last chapter come to being **fc**-multicategories. Consider, for example, the **fc**-graph MAT of matrix components (definition 3.3.1). At first sight, this appears to have a natural composition making it an **fc**-multicategory, with the composite of the diagram



given by the sequence $(x_i)_{i\in[0,\sum_{j\in[n]} r_j]}$.

However, consider the diagram

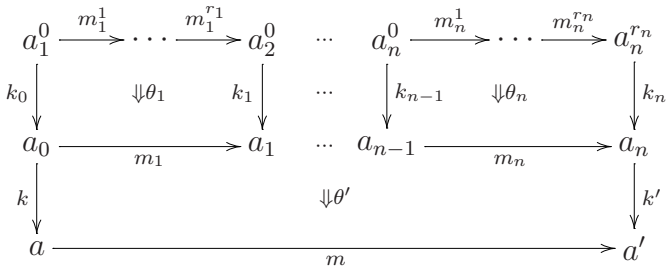


A composite of this diagram would consist of a 1-term sequence (y) with $y \in X_1^0$ and $f_0(y) = x$. There may be many such y, or there may be none at all. In any case, there is no straightforward way to define this composite.

A similar phenomenon occurs for other playpens considered in chapter 3. There is a natural composition for almost all diagrams, but not for diagrams of the shape considered in the last paragraph. Of course, no composites are needed to make the exponentials **fc**-multicategories, but some composites are needed to make these exponentials representable. As outlined in the

last section, representability involves the existence of cells with particular universal properties, whose left and right edges are identities. Accordingly, it is only composites of cells whose left and right edges are identities which we will need for these purposes. To be more precise,

Definition 4.2.1. *A diagram*



in a playpen B has a composite *iff there is a unique cell (called the composite of the diagram) which unwires to it.*

Definition 4.2.2. *A playpen B* has enough composites *iff every diagram in B in which all vertical 1-cells are identities has a composite.*

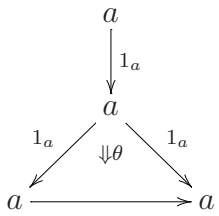
However, the property of having enough composites is not preserved by gluing, so we need to consider the following stronger condition:

Definition 4.2.3. *A playpen B* has almost all composites *iff every diagram in B in which the bottom cell has arity at least 1 has a composite.*

Example 4.2.4. mat has almost all composites, as discussed above, and ring, which is an unwirable multicategory, has almost all composites.

Lemma 4.2.5. *Every playpen with almost all composites has enough composites.*

Proof. It suffices to show that each diagram



has a composite - but of course the composite of such a diagram is θ itself. □

LEMMA 4.2.6. *Any limit of playpens with almost all (resp. enough) composites has almost all (resp. enough) composites.* □

LEMMA 4.2.7. *Let $l\colon A \to B$ be a map of playpens, where the playpens A and B each have almost all composites. Then $\mathbf{Gl}(l)$ has almost all composites.*

Proof. Given a 2-cell (θ, ρ) unwiring to a diagram D in $\mathbf{Gl}(l)$ in which the bottom cell has arity at least 1, the construction in definition 3.2.7 shows that θ and ρ decompose in A and B respectively to give diagrams determined by D and in which the bottom cells have arity at least 1. Thust (θ, ρ) is also uniquely determined by D. □

In particular, it follows that the playpen $\mathbf{Gl}(\text{ALT})$ of section 3.4 has almost all (and so enough) composites.

4.3 Weakly opcartesian cells in powers of 2

Let B be any playpen. Thus we can form exponentials of $(\mathbf{fc}/I_*V_*B)$-multicategories with respect to B. In particular, we can exponentiate (a representative of) the 2-point lattice by B. If the vertical discretisation of B is unwirable then these exponentials turn out to be representable.

DEFINITION 4.3.1. *Let B be a playpen. Then 2^B is defined to be $(C_*\mathbf{2})^B$.*

EXAMPLE 4.3.2. $2^{\mathrm{MAT}} = \mathbf{Mat}(\mathbf{2}) = \mathbf{Rel}$, the **fc**-multicategory of sets and relations.

EXAMPLE 4.3.3. The **fc**-multicategory **DigGam** of digraph games was defined as $2^{\mathbf{Gl}(\mathrm{ALT})}$ in section 3.4.

Each of C_* and $-^B$ has a left adjoint. Composing these adjunctions, we get the following universal property:

PROPOSITION 4.3.4. *Let B be a playpen. There is a map of* **fc***-graphs* ev *from $2^B \times_{V_*B} B$ to 2 such that for each* **fc** *graph X over V_*B the map from* **fc-Gph**$/V_*B(X, 2^B)$ *to* **fc-Gph**$(X \times_{V_*B} B, 2)$ *sending f to* $\mathrm{ev} \cdot (f \times_{V_*B} B)$ *is an isomorphism.* □

COROLLARY 4.3.5. *Let B be a playpen and $C \xrightarrow{f} V_*B$ a functor. Then*

$$2^{I_*C \times_{I_*V_*B} B} \cong I_*C \times_{I_*V_*B} 2^B .$$

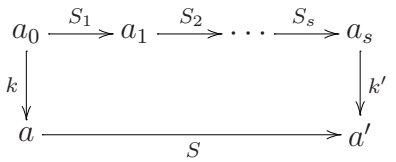
Proof. This follows from proposition 4.3.4, since for any X

$$X \times_{I_*C} (I_*C \times_{I_*V_*B} B) \cong X \times_{I_*V_*B} B .$$

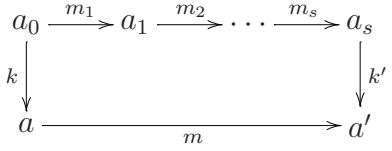
□

From now on, we shall also assume B has enough composites.

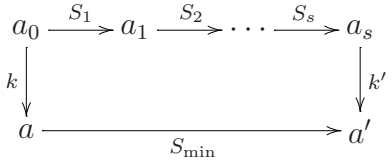
Since **2** itself is locally ordered, so is each **fc**-multicategory 2^B. The underlying category of 2^B will be V_*B. The set $2^B(a, a')$ of horizontal 1-cells from a to a' is given by the set of maps from $B(a, a')$ to $\{0, 1\}$, which we can identify with the power set of $B(a, a')$. The partial order on this given by the 2-cells of 2^B satisfies $S \leq T$ iff for all 2-cells in B with top edge in S (and both sides identities) the target is in T. In particular, this extends the order $\subseteq$. Similarly, there is a 2-cell filling the rectangle



iff for each 2-cell filling a rectangle



in B with each $m_i \in S_i$, we have $m \in S$. For a given upper edge, the set of possible lower edges S for which this holds evidently has least element $S_{\min}$ given by the set of $m \in B(a, a')$ for which there is such a 2-cell. Then the unique 2-cell

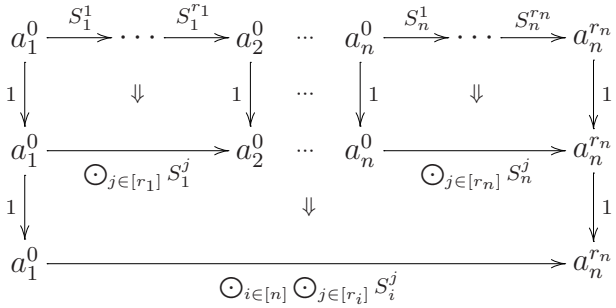


is weakly opcartesian, and so $S_{\min}$ is a pre-composite of the sequence $(S_i)_{i\in[n]}$.

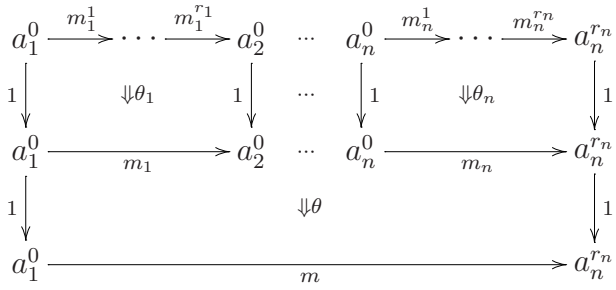
THEOREM 4.3.6. *Let B be a playpen with enough composites. Then 2^B is representable.*

Proof. The remarks above show that any composable sequence $(S_i)_{i\in[n]}$ of horizontal 1-cells in B has a pre-composite $\bigodot_{i\in[n]} S_i$. By the remarks in section 4.1, it is therefore sufficient to show that the weakly opcartesian cells constructed above are closed under composition.

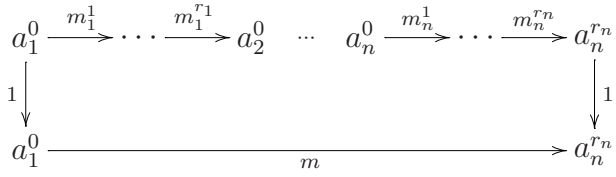
Given a composable collection K as in the diagram



of weakly opcartesian cells, the set $\bigodot_{i\in[n]} \bigodot_{j\in[r_i]} S_i^j$ is the set of lower edges m of composable collections



in B with $m_i^j \in S_i^j$ for all i and j. Since B has enough composites, these are in bijection with single 2-cells filling rectangles



with the m_j^i satisfying the same condition, and so for any $m \in B(a_1^0, a_n^{r_n})$ there is such a 2-cell with bottom edge m iff $m \in \bigodot_{i\in[n]} \bigodot_{j\in[r_i]} S_i^j$.

Thus the composite of the composable collection K is weakly opcartesian, as required. □

Example 4.3.7. Applying this to MAT, we obtain that **Rel** is representable. For a sequence $(R_i)_{i\in[n]}$ of relations, the composite is the set of pairs (x, x') for which there is a sequence $(x_i)_{i\in[0,n]}$ such that $x = x_0$, $x' = x_n$ and $(x_{i-1}, x_i) \in R_i$ for $i \in [n]$; that is, this construction recovers the usual composition of relations. In particular, the unit at X is the set of pairs $(x, x') \in X \times X$ for which there is (x_0) with $x = x_0$ and $x' = x_0$, so that $x = x'$; that is, it is the identity relation on X.

Example 4.3.8. In a similar way, **DigGam** is representable. Recalling the language of section 3.4, the composite $\tau \cdot \sigma$ of a pair of horizontal maps

$G \xrightarrow{\sigma} H \xrightarrow{\tau} K$ is given by the set of plays p in $G \Rightarrow H$ for which there is a simulation ρ in $G \otimes H \otimes K$ which follows σ and τ and whose restriction to $G \otimes K$ is p. Intuitively, it is the set of plays which could occur in $G \otimes K$ if an additional copy of H is imagined and play proceeds according to σ in $G \otimes H$ and according to τ in $H \otimes K$.

The unit at a game G is given by the set of copycat plays in $G \Rightarrow G$. Intuitively, this is the strategy in which blue always copies red's moves but in the opposite component.

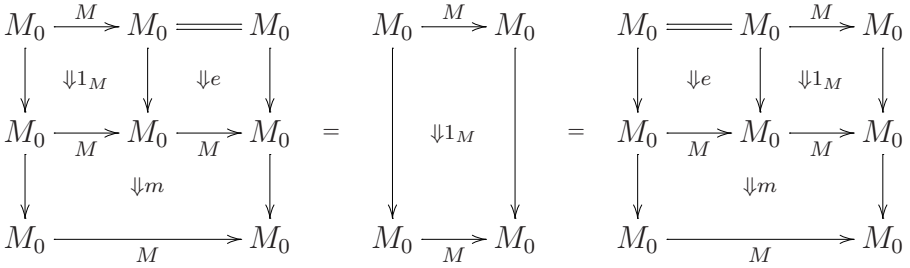
These composites and units give the horizontal digraph of **DigGam** the structure of a category, of the kind suggested in the introduction. For example, the identity at a game G is given by the set of copycat plays (see section 3.4) in $G \Rightarrow G$ - this set is called the *copycat strategy* at G.

4.4 A link to slice constructions

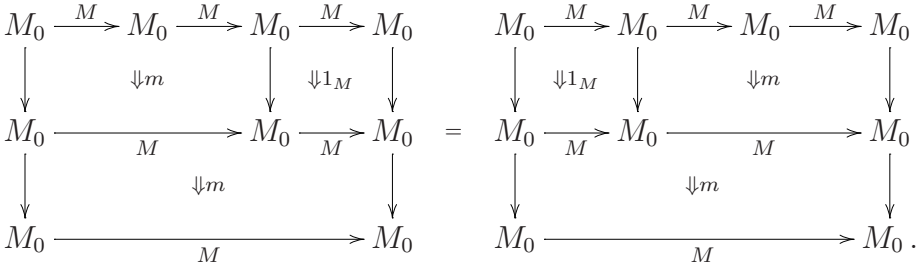
The construction as described so far has an interesting relationship to the slice construction on **fc**-multicategories. We'll begin by explaining the construction, which is reminiscent of the usual slice construction on categories. However, whilst standard slicing is performed with respect to a single object of the category being sliced, slicing for **fc**-multicategories is performed with respect to a more complicated structure, called a *horizontal monoid*.

A horizontal monoid in an **fc**-multicategory C can be thought of as a map from the terminal **fc**-multicategory to C. The terminal **fc**-multicategory consists of just one object $*$, one vertical 1-cell 1_* (the identity on that object), one horizontal 1-cell m, and, for each nonnegative integer k, one 2-cell $*_k$ from k copies of m to m. So given a multicategory C, a map from 1 to C consists of an object M_0 of C, a horizontal 1-cell $M\colon M_0 \to M_0$, and, for each nonnegative integer k, a 2-cell m_k from k copies of M to M such that the cells m_k are closed under composition and $m_1 = 1_M$. Since the 2-cells

of the terminal **fc**-multicategory are generated by $*_0$ and $*_2$, it is enough to specify two 2-cells $e = m_0$ and $m = m_2$. As is usual in such situations, it is then sufficient for m and e to satisfy the composition laws



and

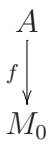


which correspond to the more familiar monoid axioms



Given an **fc**-multicategory C and a horizontal monoid M in C as above, the slice of C by that monoid is given by gluing (of multicategories) along the map $1 \to C$ corresponding to M. Explicitly, it has

- 0-cells given by vertical maps



of C

- vertical 1-cells from f to g given by vertical 1-cells h of C with $g \cdot h = f$
- vertical composites and identities given by those of C, so that the vertical category structure is given by the slice of the vertical structure of C by M_0
- horizontal 1-cells from f to f' given by 2-cells

$$\begin{array}{ccc} A & \xrightarrow{m} & A' \\ {\scriptstyle f}\downarrow & \Downarrow\theta & \downarrow{\scriptstyle f'} \\ M_0 & \xrightarrow[M]{} & M_0 \end{array}$$

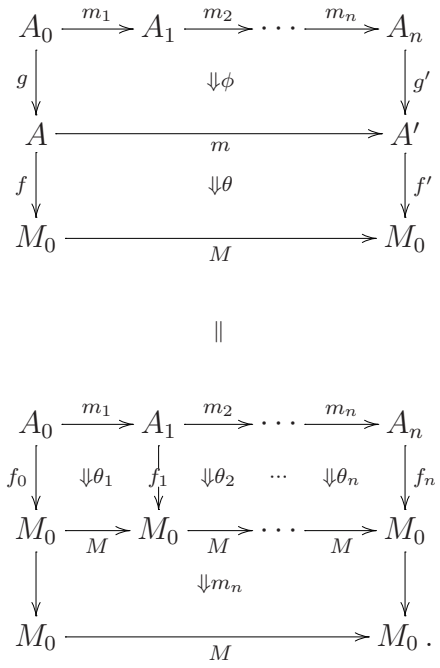
  of C
- 2-cells filling the rectangle

$$\begin{array}{ccccccc} f_0 & \xrightarrow{\theta_1} & f_1 & \xrightarrow{\theta_2} & \cdots & \xrightarrow{\theta_n} & f_n \\ {\scriptstyle g}\downarrow & & & & & & \downarrow{\scriptstyle g'} \\ f & & & & \xrightarrow[\theta]{} & & f' \end{array}$$

  given by 2-cells ϕ filling the rectangle

$$\begin{array}{ccccccc} A_0 & \xrightarrow{m_1} & A_1 & \xrightarrow{m_2} & \cdots & \xrightarrow{m_n} & A_n \\ {\scriptstyle f}\downarrow & & & & & & \downarrow{\scriptstyle f'} \\ A & & & & \xrightarrow[m]{} & & A' \end{array}$$

  in C such that



- composite and identity 2-cells given by those of C.

If C is locally ordered, the commutative diagrams of 2-cells are automatically satisfied so may be ignored.

There is a similar notion of the slice of a weak double category by a horizontal monoid. Indeed, the notions coincide if weak double categories are considered as representable **fc**-multicategories. In particular, slices of representable **fc**-multicategories are again representable. Similarly, slices of locally ordered **fc**-multicategories are again locally ordered. This suggests that slices of **fc**-multicategories produced by exponentiation may themselves sometimes be producible in this way.

To see that this is the case, first we should consider what horizontal monoids in **fc**-multicategories of the form 2^B, as considered in the last section, look like. M_0 will be an object of B, but M will be a set of horizontal 1-cells from M_0 to M_0 in B. There will be some (necessarily unique) choice of m

and e forming a horizontal monoid in this situation if and only if every 2-cell with all the 1-cells along its top edge in M and with identities down the sides also has bottom edge in M.

DEFINITION 4.4.1. *A horizontal monoid M in 2^B is* total *iff every 2-cell with bottom edge in M and identities down the sides also has all the 1-cells along its top edge in M.*

DEFINITION 4.4.2. *In this context, the* structuring **fm**-graph $\mathbf{Str}(M)$ *of M has*

- *objects given by the horizontal 1-cells contained in M*
- *2-cells given by the 2-cells of B with all the 1-cells along their top edges in M and with identities down the sides.*

$\mathbf{Str}(M)$ corresponds to a sub-playpen of B with just 1 object and 1 vertical 1-cell. Thus $\mathbf{Str}(M)$ has an unwiring inherited from that of B. If B has enough composites, this unwiring is an isomorphism, giving $\mathbf{Str}(M)$ the structure of an unwirable multicategory.

Let $\mathbf{Gl}(M)$ be the **fc**-multicategory obtained by gluing along the injection of $\mathbf{Str}(M)$ into B. $2^{\mathbf{Gl}(M)}$ isn't quite the same **fc**-multicategory as $2^B/M$. The object with respect to which we must exponentiate is not too much more complex, though.

DEFINITION 4.4.3. *In the context of the last few paragraphs, the playpen B/M is the image of the map from $\mathbf{Gl}(M)$ to the pullback of the cospan consisting of the lower and right sides of the commutative square*

$$\begin{array}{ccc} \mathbf{Gl}(M) & \longrightarrow & B \\ \downarrow & & \downarrow \\ I_*(V_*B/M_0) & \longrightarrow & I_*V_*B\,. \end{array}$$

.

PROPOSITION 4.4.4. *If B has almost all composites, and M is a total horizontal monoid in 2^B, then $2^{B/M} \cong 2^B/M$.*

Proof. The category of objects and vertical 1-cells of each of these is isomorphic to B/M_0. Horizontal 1-cells $f \to f'$ in B/M are given by horizontal 1-cells $s(f) \xrightarrow{m_1} s(f')$ in B such that there is a 2-cell filling some square

$$\begin{array}{ccc} s(f) & \xrightarrow{m_1} & s(f') \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f'} \\ M_0 & \xrightarrow[m]{} & M_0 \end{array}$$

with $m \in M$. Horizontal 1-cells $f \to f'$ in $2^{B/M}$ are given by sets of such things, that is, by sets M_1 of horizontal 1-cells $s(f) \xrightarrow{m_1} s(f')$ in B such that there is a 2-cell filling the square

$$\begin{array}{ccc} s(f) & \xrightarrow{M_1} & s(f') \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f'} \\ M_0 & \xrightarrow[M]{} & M_0 \end{array}$$

in 2^B, which are horizontal 1-cells with the same source and target in $2^B/M$.

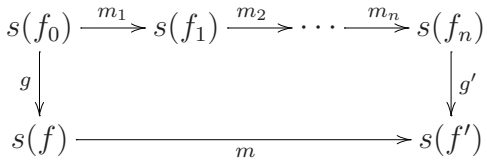
There is a 2-cell filling the rectangle

$$\begin{array}{ccccccc} f_0 & \xrightarrow{M_1} & f_1 & \xrightarrow{M_2} & \cdots & \xrightarrow{M_n} & f_n \\ {\scriptstyle g}\downarrow & & & & & & \downarrow{\scriptstyle g'} \\ f' & & & & \xrightarrow[M]{} & & f' \end{array}$$

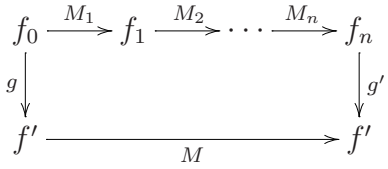
in $2^{B/M}$ iff for each 2-cell filling a rectangle

$$\begin{array}{ccccccc} f_0 & \xrightarrow{m_1} & f_1 & \xrightarrow{m_2} & \cdots & \xrightarrow{m_n} & f_n \\ {\scriptstyle g}\downarrow & & & & & & \downarrow{\scriptstyle g'} \\ f & & & & \xrightarrow[m]{} & & f' \end{array}$$

in $\mathbf{Gl}(M)$ with the $m_i \in M_i$ we have $m \in M'$. Since B has almost all composites, such 2-cells correspond to 2-cells filling rectangles



in B. So this condition is equivalent to the existence of a 2-cell



in $2^B/M$ □

Chapter 5

Basic examples and extensions

In this chapter, we'll explain some ways to use the constructions of the last few chapters to produce some familiar categories of games. So far, we've only given one construction of a category of games; namely the construction of **DigGam** which was used as a running example. The constructions in this chapter will be variations on this theme. Since the construction of **DigGam** was spread out over a few different sections, before explaining how to modify it we'll give a condensed summary of the whole construction. There are several components, which are combined according to a particular recipe, making use of the constructions which can be performed on playpens and maps of playpens.

The vertical category The first component is a category $\mathbf{Dig}_*$, together with a functor **Play** from that category to **Set** and an object of that category, in this case the digraph $R_1 =$  . The slice of $\mathbf{Dig}_*$ by R_1 will, in the end, be the vertical category of **DigGam**. $\mathbf{Dig}_*$ is the category of pointed digraphs. Digraphs were introduced in section 1.1, and the functor **Play** was introduced in section 1.3. It sends a pointed digraph to the set of finite paths starting at the basepoint and following the edges. The slice category $\mathbf{Dig}_*/R_1$ was introduced in section 3.4.

It is the category of bicoloured digraphs: pointed digraphs in which each edge has been assigned one of two colours.

The unwirable plain multicategory The second component is given by an unwirable plain multicategory RING. The construction of RING is given in section 3.3. The cells correspond to plays in the ring digraphs R_n, which encode the idea of play in games where a ring is moved around on a loop of string with knots in it, and is pulled past one such knot on each turn. This plain multicategory can be considered as an (unwirable) **fc**-multicategory with trivial vertical category, and so as a playpen.

The map to MAT The third component is a map from this playpen to MAT which sends the unique object to $\mathbf{Play}(R_1)$, the result of applying the functor from the first component to the object from the first component. This is the map KNOT introduced in section 3.4. It sends a play in R_n to the list of plays in R_1 given by considering the movements of the ring past each knot individually.

Putting it all together Since KNOT sends the unique object of RING to the set $\mathbf{Play}(R_1)$, we get as in section 3.4 a commutative square

$$\begin{array}{ccc} \mathrm{RING} & \xrightarrow{\mathrm{KNOT}} & \mathrm{MAT} \\ {\scriptstyle \overline{R_1}}\downarrow & & \downarrow{\scriptstyle \eta} \\ I_*\mathbf{Dig}_* & \xrightarrow[I_*\mathbf{Play}]{} & I_*\mathbf{Set}\,. \end{array}$$

There is an induced map of playpens from the top left corner to the pullback of the lower right cospan. Gluing along this map gives a new playpen, and raising 2 to the power of this playpen gives an **fc**-graph **DigGam**. This part of the construction is described in section 3.4.

Since the playpens at the corners of the commutative diagram above all have almost all composites (definition 4.2.3), so does the playpen we

exponentiated to obtain **DigGam**. So by theorem 4.3.6, **DigGam** is representable and can be considered as a weak double category. Since it is also locally ordered, it can be considered as a strict double category. The horizontal category (also called **DigGam**) is the final category of games produced by this construction.

In this chapter, we'll consider three kinds of modification of this construction, which are orthogonal to one another, in that modifications of the various kinds can be made independently of each other.

The first kind of modification, examined in section 5.1, focuses on the second and third components of the construction outlined above, though modifications of the third component go along with slight modifications of the first component: only the object which is sliced by will be changed. Modifications of this kind have a close link to slice constructions for **fc**-multicategories.

The second kind of modification, examined in section 5.2, takes place after the construction outlined above is complete. Tweaking the **fc**-multicategory **DigGam** by, for example, only including cells with certain closure properties, gives slight alterations of the notion of strategy. This is useful since the very general notion of strategy which appears in **DigGam** is rather unorthodox.

The third kind of modification, examined in section 5.3, focuses on the first component of the construction outlined above. We get analogous constructions with trees or recursive structures in place of digraphs.

None of the constructions of this chapter seriously adjusts the recipe outlined above. However, since playpens can be combined in a wide variety of ways, there is no reason why this recipe should always be followed. It is quite possible that for the construction of certain games-like categories we will wish to combine ingredients in more complex or radically different ways.

5.1 Modifying RING or KNOT

All the examples in this section will closely follow the recipe used to produce **DigGam**. Therefore all the **fc**-multicategories produced will be representable, and so once they have been constructed we shall not need to worry about whether we can sensibly compose the horizontal 1-cells. The vertical categories will always be based on $\mathbf{Dig}_*$. More precisely, in each case the vertical category of the **fc**-multicategory produced will be a slice of $\mathbf{Dig}_*$ by some pointed digraph, but this pointed digraph will not always be R_1. The main change to the construction will be the replacement of RING by a different unwirable plain multicategory, and so of KNOT by a different map of playpens.

5.1.1 Plays of even length

One obvious modification is to restrict to a subplaypen of RING. RING evidently has many sub-**fc**-multicategories. For example, for any subset S of the vertices, we could consider the full sub-**fc**-multicategory $\text{RING}\restriction_S$ on S. However, many of these will not be subplaypens, since the unwiring of RING may not restrict to them; there may be some cell all of whose inputs and outputs are in S but which has a decomposition containing cells without this property. There is one simple subset e on which the unwiring is preserved; the set of plays in R_2 of even length. A play in R_2 has even length iff it doesn't end up at 1. So the inputs of a cell c in RING all have even length iff c doesn't end up at i for $i \in [n]$ iff c ends up at 0 iff the output of c has even length. We note in passing that if the output of c has odd length then c must end up at some nonzero i, and that in that case all but the i^{th} input of c have even length, and the i^{th} input has odd length. In particular, all the cells in any decomposition of a cell in RING_e are still in RING_e, making RING_e unwirable and so a subplaypen of RING. This allows us to build a new

category of games, $\mathbf{DigGam}_e$.

The vertical category The first component once more consists of the category $\mathbf{Dig}_*$, together with the functor **Play** and the pointed digraph R_1. So, just like for **DigGam**, the vertical category of $\mathbf{DigGam}_e$ will be given by the category $\mathbf{Dig}_*/R_1$ of bicoloured digraphs.

The unwirable plain multicategory The second component is the full submulticategory $\textsc{ring}_e$ of RING on vertices which are plays of even length. The argument at the start of this subsection shows that $\textsc{ring}_e$ is unwirable. As before, this plain multicategory can be considered as an (unwirable) **fc**-multicategory with trivial vertical category, and so as a playpen.

The map to MAT The third component is the map $\textsc{knot}_e \colon \textsc{ring}_e \to \textsc{mat}$ given by taking the restriction of KNOT to $\textsc{ring}_e$.

Putting it all together Since $\textsc{knot}_e$ sends the unique object of $\textsc{ring}_e$ to $\mathbf{Play}(R_1)$, we get a commutative square

$$\begin{array}{ccc} \mathrm{RING}_e & \xrightarrow{\mathrm{KNOT}_e} & \mathrm{MAT} \\ {\scriptstyle \overline{R_1}}\downarrow & & \downarrow{\scriptstyle \eta} \\ I_*\mathbf{Dig}_* & \xrightarrow[I_*\mathbf{Play}]{} & I_*\mathbf{Set}\,. \end{array}$$

There is an induced map of playpens from the top left corner to the pullback of the lower right cospan. Gluing along this map gives a new playpen, and raising 2 to the power of this playpen gives an **fc**-graph $\mathbf{DigGam}_e$.

As in the case of **DigGam**, $\mathbf{DigGam}_e$ is representable and locally ordered and so can be considered as a strict double category. The horizontal category (also called $\mathbf{DigGam}_e$) is the subcategory of **DigGam** whose maps are strategies containing only plays of even length.

5.1.2 Impartial games

A similar modification involves keeping RING fixed but modifying the map KNOT. One such construction gives the category $\mathbf{DigGam}_i$ of *impartial* games.

The vertical category Just like the constructions of each of **DigGam** and $\mathbf{DigGam}_e$, we use $\mathbf{Dig}_*$ and **Play**. However, the pointed digraph we use is not R_1 but the terminal pointed digraph $1 =$  . Thus the vertical category of $\mathbf{DigGam}_i$ will be $\mathbf{Dig}_*/1 \cong \mathbf{Dig}_*$.

The unwirable plain multicategory We use the unwirable plain multicategory RING.

The map to MAT There's a unique map ! in $\mathbf{Dig}_*$ from R_1 to the the terminal digraph 1. We could 'postcompose' this with KNOT to give a map $\text{KNOT}_i\colon \text{RING} \to \text{MAT}$ which

- sends the unique object to $\mathbf{Play}(1) = \mathbb{N}_0$ and the unique map to the identity on $\mathbb{N}_0$.
- sends the horizontal 1-cell p to $\mathbf{Play}(!)^{\times 2}(\text{KNOT}(p))$.
- sends the n-ary 2-cell c to $\mathbf{Play}(!)^{\times(n+1)}(\text{KNOT}(c))$.

In other words, KNOT_i sends a typical play in the n^{th} string game to the tuple listing the numbers of times the ring passes each knot.

Putting it all together As before, we get an induced map ALT_i from RING to the pullback of η_{MAT} against $I_*\mathbf{Play}$, and so we get the **fc**-multicategory $\mathbf{DigGam}_i$ as $2^{\mathbf{Gl}(\text{ALT}_i)}$.

We've called these games impartial games because there is no indication of which moves may be made by which player. There is another way to think about impartial games.

PROPOSITION 5.1.1. *There is a pullback*

$$\begin{array}{ccc} \mathbf{DigGam}_i & \longrightarrow & \mathbf{DigGam} \\ {\scriptstyle\eta}\downarrow & & \downarrow{\scriptstyle\eta} \\ I_*\mathbf{Dig}_* & \xrightarrow[I_*(R_0^*)]{} & I_*(\mathbf{Dig}_*/R_0)\,. \end{array}$$

Proof. By corollary 4.3.5, it suffices to show that

$$\begin{array}{ccc} \mathbf{Gl}(\text{ALT}_i) & \longrightarrow & \mathbf{Gl}(\text{ALT}) \\ {\scriptstyle\eta}\downarrow & & \downarrow{\scriptstyle\eta} \\ I_*\mathbf{Dig}_* & \xrightarrow[I_*(R_0^*)]{} & I_*(\mathbf{Dig}_*/R_0)\,. \end{array}$$

is a pullback.

It is a pullback on objects and vertical 1-cells. A typical horizontal 1-cell in $\mathbf{Gl}(\text{ALT})$ from $D \times R_1$ to $D' \times R_1$ consists of plays in each of $D \times R_1$ and $D' \times R_1$ and a play in R_2. But a play in $D \times R_1$ is the same as a pair of plays, one each from D and R_1, of the same length, and similarly for D'. So a 1-cell is given by plays in each of D and D', a pair of plays in R_1, and a play in R_2 which determines those two plays in R_1. That is, it is given by the same data as a horizontal 1-cell from D to D' in $\mathbf{Gl}(\text{ALT}_i)$. The argument for 2-cells is similar. □

What this proposition shows is that $\mathbf{DigGam}_i$ is isomorphic to the subcategory of **DigGam** on games in which, although at any point the moves available to Red are different to those for Blue, they are in an exact correspondence. This isomorphism picks out the sense in which we think of impartial games as games.

5.1.3 N-coloured games and Q-coloured games

We may take the modifications of the map KNOT still further.

The vertical category We continue to use $\mathbf{Dig}_*$ and $\mathbf{Play}$. The pointed digraph N we use now is given by $U(1)$ (example 1.3.9). Recall that N has vertex set $\mathbb{N}_0$, with basepoint 0 and with a unique edge from n to $n+1$ for each n.

The unwirable plain multicategory We once again use the unwirable plain multicategory RING.

The map to MAT Recall that the map $\mathbf{Play}(!_N)\colon \mathbf{Play}(N) \to \mathbf{Play}(1)$ is a bijection. Call the inverse of this bijection u. We may, as before, 'postcompose' KNOT_i with u to get a map KNOT_n of playpens from RING to MAT, sending

- the unique object to $\mathbf{Play}(N) = \mathbb{N}_0$ and the unique map to the identity on $\mathbb{N}_0$.
- the horizontal 1-cell p to $u^{\times 2}(\text{KNOT}_i(p))$.
- the n-ary 2-cell c to $u^{\times n+1}(\text{KNOT}_i(c))$. All we have used to produce this map is that $\mathbf{Play}(!_N)$ is a bijection.

Putting it all together As before, we get an induced map ALT_n from RING to the pullback of η_{MAT} against $I_*\mathbf{Play}$, and we get the **fc**-multicategory $\mathbf{DigGam}_n$ of N-coloured games as $2^{\mathbf{Gl}(\text{ALT}_n)}$.

We don't get anything particularly new by doing this. The objects are N-coloured digraphs - these can be thought of as impartial games which keep track of how many moves have been played. In fact, since N is subterminal the **fc**-multicategory $\mathbf{DigGam}_n$ is precisely the subcategory of $\mathbf{DigGam}_i$ on digraphs with a map to N.

Rather than using N in this example, we could have used any digraph D such that the map $\mathbf{Play}(!_D)$ is an isomorphism. For example, we could have used the digraph $Q = \ 0 \rightleftarrows 1$ of section 3.4 to build the **fc**-multicategory

$\mathbf{DigGam}_p$. This construction, whilst it doesn't correspond to a simple intuition (the nearest is impartial games which keep track of the parity of the number of moves played), is interesting for the way that the structure of RING is reflected in the plays which arise. Consider, for example, simulations in the game $Q^{\otimes n+1}$ in $\mathbf{Gl}(\text{ALT}_p)$. Intuitively, each time the ring passes a knot in the underlying string game, the position in the copy of Q over that knot changes parity. If the ring crosses a knot then moves back, the parity at that knot is unchanged. So at any time the parities at each knot on one side of the ring are all the same and opposite to those on the other side of the ring. More formally, it is possible to show by induction that simulations are precisely those plays that stay in the full subgame K_{n+1} of $Q^{\otimes n+1}$ on positions consisting of a list of 0s followed by a list of 1s or a list of 1s followed by a list of 0s. This subgame is isomorphic to $R_{2(n+1)}$. Indeed, this can be used to give an alternative definition of simulations in $R_1^{\otimes n+1}$ - they are given by those plays which, when transferred to $Q^{\otimes n+1}$ in this way, remain in the corresponding subgame.

5.1.4 Replacing RING with LINE

The discussion of subsection 5.1.3 suggests an evident subplaypen of the playpen $\mathbf{Gl}(\text{ALT}_p)$. Each game K_n has a subgame L_n given by the full subgame on positions consisting of a list of 0s followed by a list of 1s. As a digraph, L_n has $n+1$ positions, which we can label with the numbers from 0 to n, and an edge from i to j iff $j = i \pm 1$. For example, L_1 is the whole of Q. If we consider only plays which remain in the L_n, we get a subplaypen of $\mathbf{Gl}(\text{ALT}_p)$, from which we can build a new **fc**-multicategory $\mathbf{DigGam}_l$ of games. We could also build $\mathbf{DigGam}_l$ (or, more precisely, an isomorphic **fc**-multicategory - we will not worry about this distinction here) in a similar way to **DigGam** but making use of another unwirable **fc**-multicategory LINE

in place of RING.

The vertical category We continue to use $\mathbf{Dig}_*$ and **Play**, but now we use the pointed digraph L_1. The vertical category of $\mathbf{DigGam}_l$ will therefore be given by $\mathbf{Dig}_*/L_1$, the objects of which are digraphs in which the vertices have been sorted into two classes and each edge links a pair of vertices of different classes.

The unwirable plain multicategory In place of the unwirable multicategory RING, we use an unwirable plain multicategory LINE is based on the line digraphs L_n in a similar manner to that in which RING is based on the R_n. Alternatively, LINE is isomorphic to the submulticategory of RING consisting of plays in which 0^+ only ever occurs as the initial move or just after 1^-, and 0^- only ever occurs just after -1^+.

The map to MAT This is the map KNOT_l built on LINE in an analogous way to the construction of KNOT on RING. Alternatively, it corresponds under the injection of LINE into RING to the restriction of KNOT to the image of this injection.

Putting it all together As before, we get an induced map ALT_l from LINE to the pullback of η_{MAT} against $I_*\mathbf{Play}$, and so we get the **fc**-multicategory $\mathbf{DigGam}_l$ as $2^{\mathbf{Gl}(\text{ALT}_l)}$.

The plays which arise in the horizontal 1-cells correspond to plays which follow a particular convention about who can play where in a combination $G \Rightarrow H$ - namely, that the opening move should be in G and each odd numbered move should be in the same component as the previous move.

5.1.5 Slicing

The constructions outlined so far have close links to the slice constructions outined in section 4.4. In particular, as explained in that section, modifica-

tions prior to exponentiation can sometimes be reinterpreted as slicing by a suitable horizontal monoid after exponentiation. To get us used to the kinds of concepts involved, let's first of all consider what slice **fc**-multicategories of **Rel** look like. **Rel** is, in a sense, the most rudimentary **fc**-multicategory of games of all. A set may be thought of as a game with the points of the set representing possible plays in the game. Under this intuition, relations are thought of as sets of plays, that is, as strategies. As 2^{MAT}, **Rel** can certainly be thought of as a simplified version of **DigGam**, and of the variations on that theme introduced earlier in this section.

A horizontal monoid M in **Rel** consists of a set M_0 and a relation M from M_0 to itself, together with a couple of 2-cells. If these 2-cells exist, they are unique, so all we need is that M should have the property that there are 2-cells filling the rectangles

$$\begin{array}{ccc} M_0 & = & M_0 \\ {\scriptstyle 1}\downarrow & & \downarrow{\scriptstyle 1} \\ M_0 & \xrightarrow[M]{} & M_0 \end{array} \qquad \text{and} \qquad \begin{array}{ccccc} M_0 & \xrightarrow{M} & M_0 & \xrightarrow{M} & M_0 \\ {\scriptstyle 1}\downarrow & & & & \downarrow{\scriptstyle 1} \\ M_0 & & \xrightarrow[M]{} & & M_0 \end{array}.$$

The first of these properties is that the identity relation on M_0 is a subset of M (so M is reflexive), and the second is that the composite of M with itself is a subset of M (so M is transitive). Thus horizontal monoids in **Rel** are given by posets.

Suppose now we have some poset $M = (M_0, \leq)$. The horizontal category of the slice of **Rel** by this poset will have objects given by M_0-coloured sets and maps from X to Y given by relations from X to Y which only ever relate points in X to points in Y with colour at least as big with respect to $\leq$. Equivalently, objects are given by M_0-tuples of sets, and maps $(X_i)_{i \in M_0} \to (Y_i)_{i \in M_0}$ by relations $X_i \to Y_j$ for $i \leq j$. Returning to the intuition above, the objects can be thought of as games in which the situations which can arise have been marked in various ways, and the maps as strategies which

obey some convention with respect to this marking.

Returning to the case of horizontal monoids in **fc**-multicategories of the form 2^B, there are some general constructions we can employ.

LEMMA 5.1.2. *Let B be a playpen with enough composites, M_0 an object of B and S a set of horizontal monoids in 2^B with object M_0. Then the intersection of all the monoids in S is again a horizontal monoid in 2^B.*

Proof. This follows from the characterisation of such monoids in section 4.4. □

EXAMPLE 5.1.3. For any object M_0 the intersection of all horizontal monoids with object M_0 in 2^B is the horizontal monoid 1_{M_0}.

EXAMPLE 5.1.4. For any object M_0 the intersection of the empty set of monoids with object M_0 in 2^B is $B(M_0, M_0)$, the set of all horizontal 1-cells from M_0 to itself in B. This, too, is always a horizontal monoid. We'll refer to the slice by it as $2^B/M_0$.

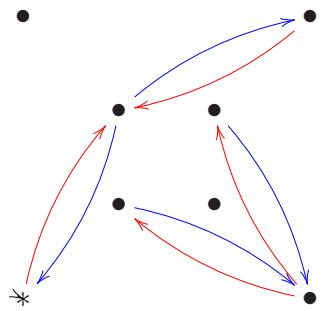
Consider, for example, the slice $\mathbf{DigGam}/1_{R_1}$ of **DigGam**. 1_{R_1} consists of all copycat plays in $R_1 \Rightarrow R_1$. These are the plays of even length in which blue always plays in the opposite component to that in which red just played. So the slice has as objects bicoloured digraphs (since R_1 is terminal in the category of bicoloured digraphs), and as maps strategies which follow this convention: they utilise only plays of even length in which blue always plays in the opposite component to that in which red just played.

That convention is a little odd. The convention of $\mathbf{DigGam}_l$ (subsection 5.1.4), which is more familiar, can also be obtained by slicing. The convention is that Red should begin by playing in the right hand game and should subsequently play in whichever game blue just played in. Orderly plays in $L_1 \Rightarrow L_1$ must always follow this convention. To visualise this fact, consider

the pullback

$$\begin{array}{ccc} M & \longrightarrow & L_1 \\ \downarrow & & \downarrow \\ L_1 \Rightarrow L_1 & \longrightarrow & R_1 \,. \end{array}$$

The pointed bicoloured digraph M is given by



in which the basepoint is $*$. The connected component of $*$ has shape given by $* \rightleftarrows \bullet \rightleftarrows \bullet$. Plays in here alternate between central positions, where blue has a choice (this is the choice of which component to play in) and outer positions, where red has no choice about the next move. The slice $\mathbf{DigGam}/L_1$ therefore has as its horizontal 1-cells the strategies only containing plays following this convention, and so it is isomorphic to $\mathbf{DigGam}_l$.

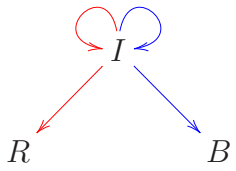
Indeed, many of the **fc**-multicategories considered so far in this section arise as slices of **DigGam** or of each other by suitable horizontal monoids. However, in some of these cases, the constructions given earlier provide either the most efficient proofs that the monoids being sliced by really are monoids or else the easiest way to understand the structure of the new sliced **fc**-multicategory. For example, $\mathbf{DigGam}_e$ is the slice of **DigGam** by the horizontal monoid with object R_1 and horizontal 1-cell given by the set of plays of even length. $\mathbf{DigGam}_i$ isn't obviously a slice of **DigGam**, but **DigGam** is the slice of $\mathbf{DigGam}_i$ by the horizontal monoid with object R_1 and horizontal 1-cell given by $\mathbf{DigGam}(R_1, R_1)$, which is the set of plays following

the playing conventions described in section 3.4. $\mathbf{DigGam}_n = \mathbf{DigGam}_i/N$ and $\mathbf{DigGam}_p = \mathbf{DigGam}_i/Q = \mathbf{DigGam}/Q$.

5.1.6 Introducing won and lost positions

There are some examples of categories of games which arise naturally from modifications of RING but which we don't know any way of producing by slicing.

The vertical category We continue to use $\mathbf{Dig}_*$ and **Play**, but now we slice by the pointed digraph Λ_1, given by



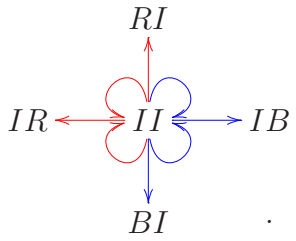
with basepoint I. Λ_1 represents the intuitive possible flow in a game in which either player may win. The game is normally 'in play' (state I), and each player may either move so as to keep the game in play or to move it to a state (R or B) in which they have won. Indeed, the intuitive idea of such games is captured by the category $\mathbf{Dig}_*/\Lambda_1$ of Λ_1-coloured games. We'll call such games *rib games*. A standard example of Λ_1-coloured games is given by the positional games considered in [6]. These are obtained by modifying the games introduced in definition 1.1.3.

DEFINITION 5.1.5. *Let (X, E) be a hypergraph (that is, $E \subseteq \mathcal{P}X$). The* positional *rib game $B(X, E)$ has as underlying digraph the subdigraph of* $\mathrm{Hyp}(X)$ *containing only positions (U, V) in which at most one of U and V extends a set in E and only edges with source (U, V) such that neither U nor V extends a set in E. Positions in which U extends a set in E have color B, those in which V extends a set in E have color*

R, and the rest have colour I. Edges adding a point to U get coloured red, and the others get coloured blue - there is only one way to colour the edges consistently with this rule.

Intuitively, in the rib game $B(X, E)$ at any stage each player has claimed a subset of X. Initially these subsets are empty. As a single move, a player may add a point which has not yet been claimed by either player to their own set of claimed points. If at any stage either player's set of claimed points extends a set in E, that player wins.

The unwirable plain multicategory In order to motivate the unwirable plain multicategory we will use, we shall first of all consider what the tensor product of rib games should look like. Taking the disjoint union of hypergraphs combines the corresponding rib games in a way which resembles the tensor product of definition 1.1.4. There is an extra clause, though - the combination is won by a player as soon as they win in one of the components. Indeed, arbitrary combinations of rib games may be formed in this way. More precisely, consider the subgame Γ of $\Lambda_1 \otimes \Lambda_1$ given by

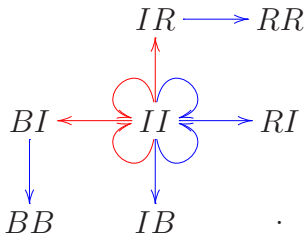


Γ has an evident rib game structure (II of colour I, RI and IR of colour R, IB and BI of colour B). We can define a tensor product on rib games by $(D, f) \otimes_\Gamma (D', f') = (f \otimes f')^{-1}\Gamma$.

We might now hope to be able to define a category of rib games using this tensor product in place of the usual tensor product of digraphs.

More specifically, a map from G to H would be a strategy in $G \otimes_\Gamma H$, following the usual play conventions captured by RING. However, this will not work in any straightforward way - for example, the copycat strategy in $(D, f) \Rightarrow (D', f')$ contains some plays which stray outside the subgame we would want to consider.

An obvious way to fix up this problem with the copycat strategies is to stipulate that, after red has made a winning move in either component, blue has just one chance to immediately make a winning move in the other component. This convention is reflected in the subgame Γ' of $\Lambda_1 \Rightarrow \Lambda_1$ given by

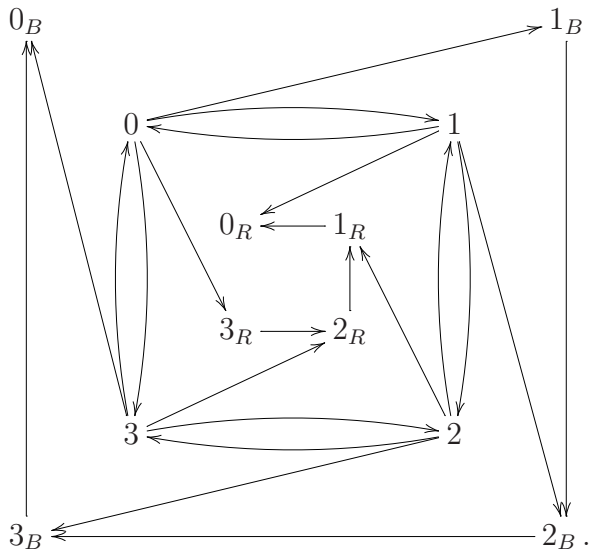


It is intuitively clear that strategies following this convention may be composed. Suppose that blue is playing the composite $g \cdot f$ of two nonlosing strategies $f\colon G \to H$ and $g\colon H \to K$, as described in the introduction. If red makes a winning move in G, f will prescribe a winning move in H, in response to which g will prescribe a winning move in K, with which blue will be able to respond. We might expect that in larger simulations wins would propogate in a similar manner either from left to right or else from right to left.

To capture this idea, we apply some basic modifications to the construction of **DigGam**. First, we must modify RING. We must extend the ring digraphs R_n to get the digraphs Λ_n which, in addition to the objects and edges of the ring digraphs, have also objects i_R and i_B for each $i \in \mathbb{Z}/n\mathbb{Z}$ and edges

- From i to $(i-1)_R$ for $i \in \mathbb{Z}/n\mathbb{Z}$.
- From i to $(i+1)_B$ for $i \in \mathbb{Z}/n\mathbb{Z}$.
- From i_R to $(i-1)_R$ for $i \in \mathbb{Z}/n\mathbb{Z} \setminus \{0\}$.
- From i_B to $(i+1)_B$ for $i \in \mathbb{Z}/n\mathbb{Z} \setminus \{0\}$.

Thus two additional spiral tracks have been added to the ring digraph, one running clockwise and the other anticlockwise, each accessible from anywhere on the central ring. For example, Λ_4 is given by



These spiral tracks correspond to the propogations of wins of the two kinds to the left or right in simulations. The constructions and arguments of section 3.3 may be carried over to this setting to give an unwirable multicategory RIB whose objects are plays in Λ_2 and whose n-ary cells are plays in Λ_{n+1}.

The map to MAT In place of KNOT, and constructed in the same manner, we have a new map SINEW from RIB (considered as a vertically trivial playpen) to MAT which sends the unique object to $\mathbf{Play}(\Lambda_1)$.

Putting it all together As before, we get an induced map WIND from RIB to the pullback of η_{MAT} against $I_*\mathbf{Play}$, and the **fc**-multicategory **RibGam** is given by $2^{\mathbf{Gl}(\mathrm{WIND})}$.

The 2-cells in $\mathbf{Gl}(\mathrm{WIND})$ are given by simulations in which, if the t^{th} move brings the i^{th} component to a position of colour B then the $(t+k)^{\mathrm{th}}$ move (if there is one) must bring the $(i+k)^{\mathrm{th}}$ component to such a position, and a symmetric condition holds for R in place of B. Thus this **fc**-multicategory corresponds to the intuition outlined in the description above.

5.2 More familiar notions of strategy

One unfamiliar thing about **DigGam** is the extremely general notion of strategy it employs for its maps. Recall from section 3.4 that a horizontal 1-cell from G to H in **DigGam** is given simply by a set of plays in a combined game $G \Rightarrow H$ which follow a standard convention. To think of these sets as strategies, they should be thought of as indicating those plays which could arise in play according to the strategy. It is more normal to work only with sets of plays (or sometimes of positions) satisfying certain conditions, which often take the form of closure properties. In this section, we'll explore how some of these conditions interact, and which combinations preserve the representability of the **fc**-multicategories involved (and so give categories of games).

We begin by considering a basic way in which the notion of strategy considered so far does not correspond to the intuitive picture. In the intuitive picture, in order to have played out some play p, it is necessary to play through all the initial segments of p. The notion of strategy considered so far fails to correspond to this because the sets of plays involved need not be closed under truncation.

One way in which we might at first attempt to resolve this is by simply imposing the condition that strategies be closed under truncation. Unfortunately, in **DigGam** the copycat strategies are not closed under truncation (they contain only plays of even length). Let $\mathbf{DigGam}_t$ be the full sub-**fc**-multicategory of **DigGam** on the truncation-closed horizontal 1-cells. We can explore the problem more carefully by analysing the representability of $\mathbf{DigGam}_t$. There is a 2-cell filling

$$\begin{array}{ccc} G & = & G \\ {\scriptstyle 1}\downarrow & & \downarrow{\scriptstyle 1} \\ G & \xrightarrow[\sigma]{} & G \end{array}$$

iff σ contains all copycat plays in $G \Rightarrow G$. There is a minimal truncation-closed horizontal 1-cell with this property; namely, the truncation closure u_G of the set of copycat plays in $G \Rightarrow G$. u_G therefore serves as the pre-unit at G in $\mathbf{DigGam}_t$. In fact, $\mathbf{DigGam}_t$ has all pre-composites. Let $(\sigma_i)_{i\in[n]}$ be a composable collection of horizontal 1-cells in $\mathbf{DigGam}_t$. Since each σ_i is truncation-closed, so is the set of simulations following the σ_i and hence so is the set of σ for which there is a simulation following all the σ_i and σ. That is, the composite of the σ_i in **DigGam** is already in $\mathbf{DigGam}_t$ and so is the composite of the σ_i there as well.

In order for it to be representable, the pre-composites of $\mathbf{DigGam}_t$ would have to be closed under composition. This fails to happen. For example, let I be the bicoloured digraph $\bullet$ and K the bicoloured digraph $\bullet \xrightarrow{i} \bullet$. Let σ be the horizontal 1-cell from I to K containing only the trivial play. Then the play consisting of just the move $(\bullet, \bullet, i)$ is a simulation following σ and u_K, so that $u_K \odot \sigma$ contains the play consisting of just the move $(\bullet, i)$. Thus $u_K \odot \sigma \neq \sigma$.

There are a couple of different approaches to resolving this, one of which is inspired by the naive understanding of what a strategy is and the other

of which (followed by most people dealing with categories of games) is more technically convenient. The two approaches give isomorphic **fc**-multicategories of games. First, the naive approach: we insist that all horizontal 1-cells σ should satisfy the condition

For each play p in σ of even length, *all* 1-move extensions of p are in σ. (♯)

This condition removes the nonrepresentability mentioned above. On its own, it fails to give a representable **fc**-multicategory, but combined with truncation closure it does give one. Let $\mathbf{DigGam}_\sharp$ be the full sub-**fc**-multicategory of **DigGam** on horizontal 1-cells satisfying (♯), and let $\mathbf{DigGam}_{t\sharp}$ be the intersection of $\mathbf{DigGam}_t$ and $\mathbf{DigGam}_\sharp$. Like $\mathbf{DigGam}_t$, $\mathbf{DigGam}_\sharp$ inherits all pre-composites other than units from **DigGam**, and it has the same units u_G as $\mathbf{DigGam}_t$. The instance of nonrepresentability sketched above is avoided since the 1-cell from I to K containing only the trivial play doesn't satisfy (♯).

A similar construction shows that $\mathbf{DigGam}_\sharp$ isn't representable. Take K as above, but now let σ be the horizontal 1-cell from K to K containing only the play $((\bullet, i), (i, \bullet))$. Then the play $((\bullet, \bullet, i), (\bullet, i, \bullet))$ is a simulation following u_K and σ, so that $\sigma \odot u_K$ contains the play $((\bullet, i))$. Thus $\sigma \odot u_K \neq \sigma$. However, in this example σ isn't truncation-closed.

The simplest way to see that $\mathbf{DigGam}_{t\sharp}$ is representable is by comparison with the more technically convenient approach. First note that each horizontal 1-cell of $\mathbf{DigGam}_{t\sharp}$ can be uniquely recovered from the set of plays of even length it contains. To put it another way, letting $U : \mathbf{DigGam} \to \mathbf{DigGam}_e$ be the map given by the identity on vertical structure and sending each horizontal 1-cell to the set of plays of even length which it contains, the restriction $U_{t\sharp}$ of U to $\mathbf{DigGam}_{t\sharp}$ is injective. The image of $U_{t\sharp}$ is given by the full sub-**fc**-multicategory $\mathbf{DigGam}_{et}$ of $\mathbf{DigGam}_e$ on horizontal 1-cells which are closed under truncation to even lengths.

The approach of only using even-length plays immediately removes the difficulty we had earlier. The units of $\mathbf{DigGam}_e$ are closed under truncation to even length, and so are also inherited by $\mathbf{DigGam}_{et}$. By the same argument as for $\mathbf{DigGam}_t$, $\mathbf{DigGam}_{et}$ inherits all pre-composites except units from $\mathbf{DigGam}_e$. Thus $\mathbf{DigGam}_{et}$ is representable. Since $U_{t\sharp}$ is an isomorphism from $\mathbf{DigGam}_{t\sharp}$ to $\mathbf{DigGam}_{et}$, $\mathbf{DigGam}_{t\sharp}$ is also representable. However, unlike $\mathbf{DigGam}_{et}$, the horizontal category of $\mathbf{DigGam}_{t\sharp}$ is not a subcategory of that of **DigGam**.

We have now got a category of games in which the strategies are closer to our intuitions. There are a few other closure conditions which are sometimes imposed and which we should consider at this point. For example, the strategies considered so far are all partial strategies. We might also want to impose a closure condition making the strategies total. This is usually formalised with the restriction that horizontal 1-cells σ should satisfy

For each play p in σ of odd length, *some* 1-move extension of p is in σ. ($\flat$)

Let $\mathbf{DigGam}_{t\sharp\flat}$ be the sub-**fc**-multicategory of $\mathbf{DigGam}_{t\sharp}$ on the horizontal 1-cells satisfying ($\flat$). The isomorphism $U_{t\sharp}$ bijects $\mathbf{DigGam}_{t\sharp\flat}$ with the full sub-**fc**-multicategory of $\mathbf{DigGam}_{et}$ on horizontal 1-cells satisfying

For each play p in σ, for *all* 1-move extensions p' of p *some* 1-move extension of p' is in σ. ($\natural$)

We'll call this sub-**fc**-multicategory $\mathbf{DigGam}_{et\natural}$, and we'll refer to its horizontal 1-cells as *continual* 1-cells.

Unfortunately, although $\mathbf{DigGam}_{et\natural}$ has units (the units of **DigGam** are continual), it is not itself representable. To see this, observe that there are continual 1-cells from I to R_1 and from R_1 to K, but not from I to K. This reflects the intuitive reason why we would not expect to get a category of games at this point. The algorithm for determining composite

strategies given in the introduction might not terminate if the middle game is not wellfounded. This algorithm was captured through the formalism of simulations, and the breakdown of the algorithm in this case can be seen in the set of simulations which follow the continual 1-cells from I to R_1 and from R_1 to K. The only such simulations are initial segments of the infinite sequence whose first term is $(\bullet, 0, i)$ with all successive terms in odd places being $(\bullet, 0^+, \bullet)$ and those in even places being $(\bullet, 0^-, \bullet)$. This infinite sequence can be thought of as an infinite computation running according to the algorithm for composition but never terminating.

A standard way to solve this translates (in this context) to considering the full sub-**fc**-multicategory $\mathbf{DigGam}_{et\natural w}$ on wellfounded games.

Definition 5.2.1. *A pointed digraph D is* wellfounded *iff the set of plays in D is wellfounded under the relation of extension.*

We shall show presently that $\mathbf{DigGam}_{et\natural w}$ is representable. However, it is possible to work in a slightly more general context, by focusing on the strategies rather than the games. The behaviour we wish to avoid in computations of the composite of $G \xrightarrow{\sigma} H \xrightarrow{\tau} K$ is play which follows both σ and τ and, after some point, remains in the game H and continues forever. We can avoid this by specifying that there should be no infinite play according to τ which, after some point, only includes moves in H.

Definition 5.2.2. *Let $G \xrightarrow{\sigma} H$ be a horizontal 1-cell of* **DigGam**. *Define the relation $\preccurlyeq_l$ on σ by $p \preccurlyeq_l q$ iff q can be obtained from p by adjoining a finite list of moves in the G component onto p. Similarly, we may define a relation $\preccurlyeq_r$ involving extension by moves in H.* σ doesn't get stuck *on the left (resp. on the right) iff it is wellfounded with respect to the reversal of $\preccurlyeq_l$ (resp. $\preccurlyeq_r$).*

Definition 5.2.3. $\mathbf{DigGam}^l$ *(resp.* $\mathbf{DigGam}^r$*) is the full sub-***fc***-multicategory of* **DigGam** *on the horizontal 1-cells which don't get stuck on the left*

(resp. on the right).

One way to think about this condition is as combining the ideas of nonlosing and winning strategy. A winning strategy in this context is a nonlosing strategy such that play is forced to terminate (since it can't terminate in a loss, it must terminate in a win). Thus the horizontal 1-cells of $\mathbf{DigGam}^l$ can be thought of as winning on the left, nonlosing on the right.

LEMMA 5.2.4. *Let $(G_i)_{i\in[0,n]}$ be digraph games, and let $(G_{i-1} \xrightarrow{\sigma_i} G_i)_{i\in[n]}$ be horizontal 1-cells between them in $\mathbf{DigGam}^l$. Let Σ be the set of simulations following all the σ_i. Define the relation $\preccurlyeq_i$ on Σ by $s \preccurlyeq_i t$ iff t can be obtained from s by adjoining a finite list of moves in the G_j components with $j < i$. Then Σ is wellfounded with respect to the reversal of $\preccurlyeq_i$ for each $i \in [0, n]$.*

Proof. By induction on i:

Base case ($i = 0$) The reversal of $\preccurlyeq_0$ is the identity relation on Σ, which is wellfounded.

Induction step Let S be any nonempty subset of Σ. Let S' be the image of S under the map $-\restriction_{\{i,i+1\}}$. Since σ_{i+1} doesn't get stuck on the left, we can find $s \in S$ such that $s\restriction_{\{i,i+1\}}$ is maximal with respect to $\preccurlyeq_l$ in S'. Let S'' be the set of simulations $\succcurlyeq_{i+1} s$ in S. By the induction hypothesis, we can find $s' \in S''$ which is maximal with respect to $\preccurlyeq_i$ in S''. Now let $t \succcurlyeq_{i+1} s'$ in S. Then $t \succcurlyeq_{i+1} s$, so $t\restriction_{\{i,i+1\}} \succcurlyeq_l s\restriction_{\{i,i+1\}}$, and so by maximality $t\restriction_{\{i,i+1\}} = s\restriction_{\{i,i+1\}}$ and in particular $t\restriction_{\{i\}} = s'\restriction_{\{i\}}$. Therefore $t \succcurlyeq_i s'$ and by maximality we have $t = s'$. That is, s' is maximal with respect to $\preccurlyeq_{i+1}$ in S, as required.

□

COROLLARY 5.2.5. $\mathbf{DigGam}^l$ *is representable.* □

We can combine the condition of not getting stuck with the others considered so far.

DEFINITION 5.2.6. *The* **fc***-multicategory* $\mathbf{DigGam}^{l}_{et\natural}$ *(resp.* $\mathbf{DigGam}^{r}_{et\natural}$*) is the full sub-*$\mathbf{fc}$*-multicategory of* $\mathbf{DigGam}_{et\natural}$ *on the horizontal 1-cells which don't get stuck on the left (resp. on the right).*

THEOREM 5.2.7. $\mathbf{DigGam}^{l}_{et\natural}$ *is representable.*

Proof. Since the units of **DigGam** are continual, it is enough to show that composites of nontrivial collections are continual. Let $(G_i)_{i\in[0,n]}$, with $n > 0$, be digraph games, and let $(G_{i-1} \xrightarrow{\sigma_i} G_i)_{i\in[n]}$ be continual 1-cells between them which don't get stuck on the left. For each i, let $\widehat{\sigma_i}$ be the truncation-closure of σ_i, so that each $\widehat{\sigma_i}$ is a horizontal 1-cell in $\mathbf{DigGam}_{t\sharp\flat}$ and doesn't get stuck on the left. Let Σ be the set of simulations following all the $\widehat{\sigma_i}$, and define the relation $\preccurlyeq_n$ on Σ as in lemma 5.2.4.

Let $p \in \bigodot_{i\in[n]} \sigma_i$, and let p' be any 1-move extension of p, so that $p' \in \bigodot_{i\in[n]} \widehat{\sigma_i}$. Let S be the set $\{s \in \Sigma | s\restriction_{\{0,n\}} = p'\}$. By lemma 5.2.4, we can find s which is maximal in S with respect to $\preccurlyeq_n$. Since the length of p' is odd, it follows from the analysis at the beginning of section 5.1.1 that there is a unique $i \in [n]$ with $s\restriction_{\{i-1,i\}}$ of odd length. There is a one move extension q of $s\restriction_{\{i-1,i\}}$ in σ_i, and a unique one-move extension s' of s with $s'\restriction_{\{i-1,i\}} = q$. s' is in Σ since each $\widehat{\sigma_j}$ has property ($\sharp$). By maximality of s, $s'\restriction_{\{0,n\}}$ is a one-move extension p'' of p'. Since p'' has even length, so does each play $s\restriction_{\{i-1,i\}}$ and so s follows the σ_i. Thus $p'' \in \bigodot_{i\in[n]} \sigma_i$, as required. □

Since no horizontal 1-cell between wellfounded games can get stuck on the left, it follows that $\mathbf{DigGam}_{et\natural w}$ is also representable. Therefore the sub-**fc**-multicategories of $\mathbf{DigGam}_{t\sharp\flat}$ corresponding to these are also representable - it is the horizontal categories of these which have what have been traditionally considered strategies as their morphisms.

5.3 Replacing $\mathbf{Dig}_*$

All the categories of games introduced so far have been built on digraphs. This is because of the flexibility of this notion of game and in particular the ability to impose standard structures by slicing by simple objects. However, the approach extends equally well to other combinatorial notions of game. In this section, we'll sketch how to deal with two other standard combinatorial approaches to games - via trees and via the recursively defined structures of Conway.

Recall that the use of 2-dimensional structures like **fc**-multicategories reflects the fact that the combinatorial structures underlying games will, as combinatorial structures, have maps of their own which form a category, in addition to the maps provided by the strategies. The vertical structure is given by the maps-as-combinatorial-objects, and the horizontal structure deals with the strategies. (A slice of) $\mathbf{Dig}_*$ provided the vertical structure for **DigGam**, so in order to implement different notions of game we shall have to replace $\mathbf{Dig}_*$ by something else.

5.3.1 Tree games

The link to trees is relatively simple - recall from section 1.3 that there is an idempotent comonad U on $\mathbf{Dig}_*$ whose category of coalgebras **Tree** is naturally interpreted as a category of trees. The functors of the induced adjunction from $\mathbf{Dig}_*$ to **Tree** are denoted d and t. U sends a pointed digraph D to the pointed digraph on $\mathbf{Play}(D)$ whose edges are given by 1-move extension. Coalgebras are given by digraphs in which there is a unique play leading to any position. Formally, this correspondence says that the functor $\mathbf{Pos} = V \cdot d \colon \mathbf{Tree} \to \mathbf{Set}$ sending a digraph to its set of positions is naturally isomorphic to $\mathbf{Play} \cdot d$. They can be implemented as diagrams of

shape

in **Set**, with positions given by the elements of the sets involved and edges given by elements of the graphs of the functions. Full details are in section 1.3.

It is usual, given a tree

$$T_1 \xleftarrow{s_1} T_2 \xleftarrow{s_2} T_3 \xleftarrow{s_3} T_4 \longleftarrow \cdots \quad ,$$

to think of the elements of odd-numbered sets as positions in which Red has just played, and the elements of even-numbered sets as positions in which blue has just played. Recalling the digraph $Q = \; 0 \rightleftarrows 1$, this thought corresponds to the fact that $t(Q)$ is terminal in **Tree**. Thus $\mathbf{Tree} \cong \mathbf{Tree}/t(Q)$, and we get a fully faithful forgetful functor $d' \colon \mathbf{Tree} \to \mathbf{Dig}_*/Q$. We can now immediately obtain an **fc**-multicategory of trees. Take **TreeGam** to be the full subcategory of $\mathbf{DigGam}_l$ on objects in the image of d'.

However, it is not necessary to go via digraphs in order to obtain this **fc**-multicategory. We may build it in a similar way to **DigGam**. There is a commutative diagram

$$\begin{array}{ccc} \textsc{line} & \longrightarrow & \textsc{mat} \\ {\scriptstyle \overline{t(Q)}}\downarrow & & \downarrow{\scriptstyle \eta} \\ I_*\mathbf{Tree} & \xrightarrow[I_*\mathbf{Pos}]{} & I_*\mathbf{Set} \end{array}$$

which induces a map $\textsc{alt}'$ from $\textsc{line}$ to P', the pullback of the lower and right edges of this square. Since **Play** sends the counit at Q to an isomorphism, there is a pullback

$$\begin{array}{ccc} \mathbf{Gl}(\textsc{alt}') & \longrightarrow & \mathbf{Gl}(\textsc{alt}_l) \\ {\scriptstyle \eta}\downarrow & & \downarrow{\scriptstyle \eta} \\ I_*\mathbf{Tree} & \xrightarrow[I_*d']{} & \mathbf{Dig}_*/Q\,, \end{array}$$

and so by corollary 4.3.5 $\mathbf{TreeGam} \cong 2^{\mathbf{Gl}(\textsc{alt}')}$.

The composition in cases like this, following the conventions of LINE rather than RING, may be (and usually is) treated from a 1-dimensional point of view. This is because LINE itself can be seen as only encoding a 1-dimensional object. It can be built in a straightforward way from a relatively simple category; the category **Sched** of *schedules*, as constructed in [7]. Let's briefly consider this fact, approaching it from the side of LINE.

Recall that LINE has only one object and one vertical map. However, we can 'thicken' LINE to a larger set of objects. Let LINE$'$ have as objects the elements of $\mathbf{Play}(L_1)$, as vertical maps only identities, and as horizontal 1-cells and 2-cells those of LINE, where each horizontal 1-cell has source and target given by its two restrictions to L_1. In the language of section 3.2, letting $\text{KNOT}_l\colon \text{LINE} \to \text{MAT}$ be the map playing the role of KNOT in the construction of $\mathbf{DigGam}_l$, LINE$'$ is the full sub-**fc**-multicategory of $\mathbf{Gl}(\text{KNOT}_l)$ on objects (a, f) where the source of f is the 1-point set $\{*\}$. The structure of LINE$'$ is essentially determined by that of **Sched**.

PROPOSITION 5.3.1. *For each composable sequence* $(a_{i-1} \xrightarrow{m_i} a_i)_{i\in[n]}$ *of horizontal 1-cells in* LINE$'$*, there is a unique 2-cell with top edge* $(m_i)_{i\in[n]}$.

Proof. Let p be a play which follows truncations of the m_i. We'll show that, unless p already follows the m_i, there is unique 1-move extension of p with the same property. Suppose first of all that the target of p is the object 0 of L_{n+1}. The only possible 1-move extension of p is that which adjoins 0^+. If this doesn't follow truncations of the m_i, p must follow m_1 and so by induction p must follow m_i and the target of m_i must be 0 for each i. In particular, p follows all the m_i. A similar argument works if p has target $n+1$. If the target of p is some other j, then if p already follows m_j a similar argument shows that it follows all the m_i. Otherwise, the next move of m_i after those so far traversed by p uniquely determines the next move.

Thus we can inductively build the unique play p following the m_i. □

In particular, LINE′ is representable and locally ordered; indeed it is completely determined by its horizontal category, which is **Sched**. There is a unique 2-cell from $(m_i)_{i\in[n]}$ to m iff m is the composite of the m_i in **Sched**. This is not the usual approach to **Sched**. Indeed, there is a simple and entirely independent combinatorial construction. Thus **Sched** may be used to found the structure of **Tree**. The formalisation of **Tree** in terms of **Sched** is, however, a recent development of what had previously been presented as a more hands-on construction.

5.3.2 Conway games

Let's now turn our attention to another familiar category of games; that of Conway.

DEFINITION 5.3.2. *A* Conway game *is a pair of families of Conway games. A* map of Conway games *from* $((g_i)_{i\in I}, (h_j)_{j\in J})$ *to* $((g'_i)_{i\in I'}, (h'_j)_{j\in J'})$ *is a pair of maps* $(k\colon I \to I', l\colon J \to J')$ *together with families of maps* $(g_i \xrightarrow{u_i} g'_{k(i)})_{i\in I}$ *and* $(h_i \xrightarrow{v_i} h'_{l(j)})_{j\in J}$ *of Conway games.*

This definition differs from that of Conway, in that Conway used sets rather than families. However, the use of families is necessary for technical reasons in order to form a category of games whose objects are Conway games. The first explicit presentation of a category of Conway games with maps given by strategies was given by Joyal in [11].

The definition above is recursive; it can be made to fit within a standard set-theoretic framework by defining games and maps of rank α for each ordinal α, but since we shall not need to worry about set-theoretic issues we shall not bother to do this here.

This recursive framework allows basic operations to be defined rapidly.

DEFINITION 5.3.3. *Let* $G = ((g_i)_{i\in I}, (h_j)_{j\in J})$ *and* $G' = ((g'_i)_{i\in I'}, (h'_j)_{j\in J'})$ *be*

Conway games.

$$G \leq G' \leftrightarrow ((\forall i \in I) g_i \not\geq G') \wedge ((\forall j \in J') G \not\geq h'_j)$$

$$-G = ((-h_j)_{j \in J}, (-g_i)_{i \in I})$$

$$G + G' = ((g_i + G')_{i \in I} \sqcup (G + g'_i)_{i \in I'}, (h_j + G')_{j \in J} \sqcup (G + h'_j)_{j \in J'})$$

A second player winning strategy *in G is a function assigning to each $i \in I$ a first player winning strategy in g_i. A* first player winning strategy *in G is a nonempty set of pairs (j, σ) with $j \in J$ and σ a second player winning strategy in h_j.*

Various properties of these operations may also be demonstrated by recursion; for example, $\leq$ is reflexive and transitive, $+$ is commutative and associative with identity $(\emptyset, \emptyset)$, where $\emptyset$ is the empty family. The preorder $\leq$ can be thickened to give a category structure. $G \leq G'$ iff there is a second player winning strategy in $G' - G$. The category **ConGam** has Conway games as objects and second player winning strategies in $G' - G$ as maps from G to G'. The composites and identities may be defined recursively.

This category can also be defined in a similar way to **DigGam**. All the structure of plays, simulations and so forth can be defined recursively. RING itself and maps from RING cannot be defined recursively since the ring digraphs aren't wellfounded. So the construction of simulations cannot be phrased in a way which mirrors that for digraph games and which remains within the recursive philosophy of Conway's approach. Instead, if we wish to employ recursion, simulations must be defined in a more hands-on way.

DEFINITION 5.3.4. *Let $G = ((g_i)_{i \in I}, (h_j)_{j \in J})$ and $G' = ((g'_i)_{i \in I'}, (h'_j)_{j \in J'})$ be Conway games. A* play *in G is either the trivial play 1_G, a pair (i, p) with $i \in I$ and p a play in g_i, or a pair (j, q) with $j \in J$ and q a play in h_j. A play p in G is* even-orderly *iff it is 1_G or (j, q) with $j \in J$ and q odd-orderly.*

A play q in G is odd-orderly *iff it is (i,p) with $i \in I$ and p even-orderly. A play in G is* orderly *iff it is either even-orderly or odd-orderly.*

Let $(G^z = ((g^z_i)_{i\in I^z}, (h^z_j)_{j\in J^z}))_{z\in[0,n]}$ be a family of Conway games. Let $f\colon G^0 \to G$ given by the maps $(k, l, (p_i)_{i\in I^0}, (q_j)_{j\in J^0})$ and $f'\colon G^n \to G'$ given by the maps $(k', l', (p'_i)_{i\in I^n}, (q'_j)_{j\in J^n})$ be maps of Conway games. Let $(p_z)_{z\in[n]}$ be orderly plays in the games $G_z - G_{z-1}$, and p an orderly play in $G' - G$. A simulation *including the p_z and p and initiated at $z_0 \in \mathbb{Z}/(n+1)\mathbb{Z}$ is either the trivial simulation 1 (if all the p_z and p are trivial) or a triple (z_1, i, s) such that one of the following holds:*

- *$z_1 \cong_{n+1} z_0 - 1$, and s is a simulation including the plays q_z and q and initiated at z_1, where $q_z = p_z$ for $z \notin z_0, z_1$, $p_{z_0} = (i, q_{z_0})$ and $p_{z_1} = (i, q_{z_1})$, and p is given by $(k(i), q)$ if z_0 is 1, by $(l'(i), q)$ if z_0 is $n+1$ and by q otherwise.*

- *$z_1 \cong_{n+1} z_0 + 1$, and s is a simulation including the plays q_z and q and initiated at z_1, where $q_z = p_z$ for $z \notin z_0, z_1$, $p_{z_0} = (i, q_{z_0})$ and $p_{z_1} = (i, q_{z_1})$, and p is given by $(k'(i), q)$ if z_0 is 0, by $(l(i), q)$ if z_0 is n and by q otherwise.*

Furthermore, with a little more recursive definition we get the structure of a playpen CON with objects given by Conway games, vertical 1-cells given by maps of Conway games, horizontal 1-cells $G \to G'$ given by orderly plays in $G' - G$ and 2-cells given by simulations initiated at 0. The horizontal category of 2^{CON} is not quite the same as **ConGam** - **ConGam** is the subcategory given by continual 1-cells, as in section 5.2.

However, it is not just true that **ConGam** can be constructed in a similar way to $\mathbf{DigGam}_{et\natural w}$. Conway games can be straightforwardly interpreted as bicoloured digraphs.

DEFINITION 5.3.5. *Let $(d_i)_{i\in I}$ and $(e_j)_{j\in J}$ be bicoloured digraphs. Then we define $[(d_i)_{i\in I}, (e_j)_{j\in J}]$ to be the bicoloured digraph with object set given by*

the disjoint unions of the object sets of the d_i and the e_j together with a new basepoint $*$, and edge set given by the disjoint union of the edge sets of the d_i and e_j together with a red edge from $*$ to the basepoint of d_i for each $i \in I$ and a blue edge from $*$ to the basepoint of e_j for each $j \in J$. Let $G = ((g_i)_{i\in I}, (h_j)_{j\in J})$ be a Conway game. Then $[\![G]\!] = [([\![g_i]\!])_{i\in I}, ([\![h_j]\!])_{j\in J}]$.

Call a digraph *accessible* if every vertex is the target of some play. It is clear by recursion that, for any Conway game G, $[\![G]\!]$ is accessible. Let f be a map of digraphs from $[(d_i)_{i\in I}, (e_j)_{j\in J}]$ to $[(d'_i)_{i\in I'}, (e'_j)_{j\in J'}]$, with the d_i and e_i accessible. f must send the basepoint of each d_i to the basepoint of some d'_i; this gives a map $k\colon I \to I'$. Further, since d_i is accessible, the map f must restrict to a map $p_i\colon d_i \to d'_{k(i)}$. Similarly, we get a map $l\colon J \to J'$ and for each $j \in J$ a map of digraphs $q_j\colon e_j \to e'_j$. Given such maps, we can reconstruct the map f. By recursion, we obtain an extension of $[\![-]\!]$ to a fully faithful functor to $\mathbf{Dig}_*/R_1$.

CON is isomorphic to the full sub-playpen of $\mathbf{Gl}$(ALT) on the image of $[\![-]\!]$, and it follows that **ConGam** is isomorphic to the full subcategory of the horizontal category of $\mathbf{DigGam}_{et\natural w}$ on the digraphs of the form $[\![G]\!]$.

Bibliography

[1] Samson Abramsky, Radha Jagadeesan, and Pasquale Malacaria. Full abstraction for PCF. *Information and Computation*, 163:409–470, 1996.

[2] F. Conduchet. Au sujet de l'existence d'adjoints à droite aux foncteurs 'image réciproque' dans la catégorie des catégories. *C. R. Acad. Sci. Paris*, (275), 1972.

[3] John H. Conway. *On Numbers and Games*. AK Peters, Ltd., 2000.

[4] G. S. H. Cruttwell and Michael A. Shulman. A unified framework for generalized multicategories, 2009.

[5] Brian Day. On closed categories of functors. In S. MacLane, editor, *Reports of the Midwest Category Seminar*, volume 137 of *Lecture Notes in Mathematics*, pages 1–38, Berlin–New York, 1970. Springer-Verlag.

[6] A. W. Hales and R. I. Jewett. Regularity and positional games. *Transactions of the American Mathematical Society*, (106), 1963.

[7] Russell Harmer, Martin Hyland, and Paul-André Melliès. Categorical combinatorics for innocent strategies. In *LICS*, pages 379–388, 2007.

[8] J. M. E. Hyland and C.-H. Luke Ong. On full abstraction for PCF: I, II, and III. *Inf. Comput.*, 163(2):285–408, 2000.

[9] M. Hyland. Game semantics. In *Semantics and Logics of computation*, Publications of the Newton Institute, pages 131–184. Cambridge University Press, 1997.

[10] Peter T. Johnstone. *Sketches of an elephant: A Topos Theory Compendium (vol. 1)*. Number 43 in Oxford Logic Guides. Oxford University Press, 2002.

[11] A. Joyal. Remarques sur la théorie des jeux a deux personnes. *Gazette des sciences mathématiques du Québec*, (1,4), 1977.

[12] G. M. Kelly. Doctrinal adjunction. In *Category Seminar*, number 420 in Lecture Notes in Mathematics, pages 257–280. Springer-Verlag, 1974.

[13] Tom Leinster. *Higher Operads, Higher Categories*. Number 298 in London Mathematical Society Lecture Note Series. Cambridge University Press, 2004.

Appendix A

Playful constructions

Composing V, the functor sending any pointed digraph to its set of vertices, with U gives the colax monoidal functor $\mathbf{Play}\colon \mathbf{Dig}_* \to \mathbf{Set}$ sending any pointed digraph to the set of plays from the basepoint in that digraph. This functor played a key role in the running example of a construction of a category of games. Its colax monoidal structure, though not essential for that construction, was helpful in giving a hands on combinatorial presentation of the structure in section 3.4. A similar phenomenon happens in a more general context, that of playful constructions, which can be considered as generalisations of cartesian colax monoidal functors to $(\mathbf{Set}, \times)$.

In section A.1 we'll pick out the properties of the functor **Play** which are necessary for the general construction to proceed and so define playful constructions. I'll also point out some basic properties. Then in section A.2 we'll outline how these constructions relate to the earlier material, and particularly to the gluing construction of section 3.2.

A.1 CCM and Pre-colax functors, and playful constructions

Recall that a category is *cartesian* iff it has pullbacks, a functor is *cartesian* iff it preserves pullbacks, a natural transformation is *cartesian* iff all the naturality squares are pullbacks, a monad is *cartesian* iff the defining functor and natural transformations are all cartesian, and so on. In the same spirit,

Definition A.1.1. *A colax monoidal functor f is* cartesian colax monoidal *(or* CCM*) iff f itself and the natural transformation $f \cdot \otimes \to \otimes \cdot (f \times f)$ are both cartesian.*

Lemma A.1.2. *Any strong monoidal functor is CCM.* □

Lemma A.1.3. *Any composite of CCM functors is CCM.* □

We'll shortly see that the functor U introduced in section 1.3 is CCM. Before showing this, however, we need a little context. For any digraph D the 'free algebra' monad T_D for the theory $\mathcal{T}_D$ (as in section 1.3) is cartesian, since that theory is strongly regular (having no equations at all). Similarly, for any pair (D_1, D_2) of digraphs, let the theory $\mathcal{T}_{D_1,D_2}$ be formed by taking the tensor product of the theories $\mathcal{T}_{D_1}$ and $\mathcal{T}_{D_2}$. Once more, $\mathcal{T}_{D_1,D_2}$ is strongly regular and so the corresponding free algebra monad T_{D_1,D_2} on $\mathbf{Set}^{V(D_1)\times V(D_2)}$ is cartesian. This construction gives a monad T on the category $\mathcal{E}$ of triples (D_1, D_2, X), where D_1 and D_2 are digraphs and $X \in \mathbf{Set}^{V(D_1)\times V(D_2)}$, sending (D_1, D_2, X) to $(D_1, D_2, T_{D_1,D_2}X)$, and once more this monad is cartesian.

$\mathcal{E}$ has a terminal element 1 consisting of two copies of the terminal digraph and the 1-point set. The first two elements of $T1$ are again each given by the terminal digraph, but the third element of $T1$ is given by the set of ordered pairs of natural numbers. The action of the unique map in the first component is to increase the first natural number by 1; the action of that

in the second component is to increase the second natural number by 1. Consider the free T-operad O on a pair of maps, one with source given by $(1,0)$ and the other with source given by $(0,1)$. An algebra for O consists of a triple (D_1, D_2, X), where D_1 and D_2 are digraphs and X is an element of $\mathbf{Set}^{V(D_1)\times V(D_2)}$ with actions of D_1 and D_2 on X; this is not the same as an algebra for $\mathcal{T}_{D_1,D_2}$, since these actions needn't commute. Instead, X is an algebra for the direct sum $\mathcal{T}'_{D_1,D_2}$ of $\mathcal{T}_{D_1}$ and $\mathcal{T}_{D_2}$. To put it another way, X is an algebra for $\mathcal{T}_{D_1\otimes D_2}$. Letting T_O be the free O-algebra monad on $\mathcal{E}$, there is as in [13, §6.2] a cartesian natural transformation $\alpha\colon T_O \to T$. That α is cartesian here could be thought of as a reflection of the fact that the free sesquicategory monad is a club over the free bicategory monad.

Proposition A.1.4. *U is CCM.*

Proof. Consider the functor K from $\mathbf{Dig}_* \times \mathbf{Dig}_*$ to $\mathcal{E}$ which sends the pair $((D_1, x_1), (D_2, x_2))$ to the triple (D_1, D_2, X) where X has a single element of type (x_1, x_2), and the cartesian functor L sending an algebra for $\mathcal{T}_{D_1\otimes D_2}$ to the corresponding digraph, as in section 1.3. Then $T_O \cdot K$ sends $(D_1, x_1), (D_2, x_2)$ to $(D_1, D_2, U((D_1, x_1)\otimes(D_2, x_2)))$, So $L\cdot T_O \cdot K$ sends (D_1, D_2) to $U(D_1\otimes D_2)$. On the other hand, letting X_1 have a single element of type x_1 and X_2 a single element of type x_2, we get $T \cdot K((D_1, x_1), (D_2, x_2)) = (D_1, D_2, T_{D_1}(X_1) \times T_{D_2}(X_2))$ so that $L \cdot T \cdot K(D_1, D_2) = U(D_1) \otimes U(D_2)$. Furthermore, the natural transformation $L \cdot \alpha \cdot K$ is the colax monoidal structure on U. Thus U is cartesian colax monoidal. □

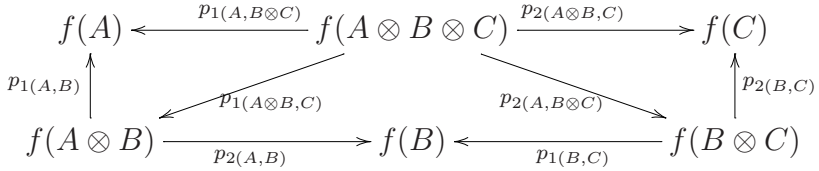
Corollary A.1.5. **Play** *is CCM.* □

This corollary can also be checked by explicit calculation of the relevant pullbacks.

The constructions of this appendix work with CCM functors to $(\mathbf{Set}, \times)$. However, they also work in a slightly more general context. To introduce

that context, we need a generalisation of the notion 'colax monoidal functor where the monoidal structure on the codomain is given by $\times$'.

Definition A.1.6. *Let C be a monoidal category, with monoidal structure given by the tensor product $\otimes$ and the structure maps λ, ρ and α. Let D be any category. A* pre-colax *functor $C \to D$ is a functor $f\colon C \to D$ together with natural transformations $p_1\colon f\cdot\otimes \to f\cdot\pi_1$ and $p_2\colon f\cdot\otimes \to f\cdot\pi_2$ such that $p_{1(-,I)} = f\rho_-$, $p_{2(I,-)} = f\lambda_-$ and the following diagram always commutes:*



Example A.1.7. Suppose that D has finite products. Then a colax monoidal structure on f from $(C,\otimes)$ to $(D,\times)$ is given by a natural transformation from $f\cdot\otimes$ to $\times\cdot(f\times f)$, satisfying certain conditions. But since $\times$ is right adjoint to Δ, this is equivalent to a natural transformation from $\Delta\cdot f\cdot\otimes$ to $f\times f$, or to a pair (p_1,p_2) of natural transformations as in the definition above. So if D has products then f is pre-colax iff it is colax monoidal from $(C,\otimes)$ to $(D,\times)$.

So a pre-colax functor $C \to D$ can be thought of as a colax monoidal functor to D with the tensor product structure on D given by taking products, even if D doesn't have products. In fact, in all the cases considered here, D *does* have products. Even so, the notion of a pre-colax functor is useful in that it gives a simple description of a universal property we shall need.

Given a category C, the construction of the free (strict) monoidal category $\mathbf{FMon}(C)$ on C is standard: The objects are finite lists of objects of C, and the maps are finite lists of maps of C. Now suppose that C is strict

monoidal, and that there is some pre-colax functor $f\colon C \to D$. Then there is a corresponding monoidal map $\mathbf{FMon}(C) \to C$ and so a pre-colax map $\mathbf{FMon}(C) \to D$. This pre-colax map factors through the free image $\mathbf{List}(C)$ of a pre-colax map from $\mathbf{FMon}(C)$, if there is such a free image.

What might such a free image $\mathbf{List}(C)$ look like? There must be an object for each object of $\mathbf{FMon}(C)$, that is, for each list of objects of C, and there must be a map for each list of maps of C. There must also be maps corresponding to those given by the natural transformations p_1 and p_2. Composing these maps, for each pair of lists $(c_i)_{i\in[m]}$ and $(d_j)_{j\in[n]}$, each injective map of ordered sets $k\colon [n] \to [m]$ and each list $(c_{k(j)} \xrightarrow{f_j} d_j)_{j\in[n]}$ of arrows of C gives an arrow in $\mathbf{List}(C)$ from $(c_i)_{i\in[m]}$ to $(d_j)_{j\in[n]}$. It is not hard to check that this is all that is needed.

To phrase it more formally, let $\mathbf{List}(C)$ be the familiar free strict monoidal category on C in which the identity is terminal. The objects and maps of $\mathbf{List}(C)$ can be given a combinatorial description as in the previous paragraph. Then $\mathbf{List}(C)$ has the universal property of being the free image of a pre-colax map from $\mathbf{FMon}(C)$.

Now any pre-colax map $f\colon C \to D$ can be extended to a pre-colax map $\mathbf{FMon}(C) \to D$ and so gives a canonical map $\overline{f}\colon \mathbf{List}(C) \to D$.

Example A.1.8. The identity functor on **Set** is colax monoidal and so pre-colax. So this construction gives a map $\mathbf{Tuple} = \overline{1}\colon \mathbf{List}(\mathbf{Set}) \to \mathbf{Set}$. This map sends a list $(X_i)_{i\in[n]}$ of sets to the set of lists $(x_i)_{i\in[n]}$ such that $x_i \in X_i$ for all i.

Example A.1.9. The map $\mathbf{Play}\colon \mathbf{Dig}_* \to \mathbf{Set}$ is colax monoidal and so pre-colax. So this construction gives a map $\overline{\mathbf{Play}}\colon \mathbf{List}(\mathbf{Dig}_*) \to \mathbf{Set}$. $\overline{\mathbf{Play}}$ sends the list $(D_i)_{i\in[n]}$ of digraphs to the set of plays in $\bigotimes_{i\in[n]} D_i$.

The construction of this appendix uses a particular property of $\overline{\mathbf{Play}}$, which follows from the fact that **Play** is CCM. The following definition is

the needed generalisation of the notion of a CCM functor:

Definition A.1.10. *Let C be a category. A* playful construction *on C consists of a functor $F\colon \mathbf{List}(C) \to \mathbf{Set}$ for which all the canonical squares*

$$\begin{array}{ccc} F((A_i)_{i\in[n]}) & \longrightarrow & F((A'_i)_{i\in[n]}) \\ \downarrow & & \downarrow \\ \prod_{i\in[n]} F(A_i) & \longrightarrow & \prod_{i\in[n]} F(A'_i) \end{array}$$

are pullbacks.

Example A.1.11. If C is a monoidal category and f is a CCM functor $C \to \mathbf{Set}$ then $\overline{f}$ is a playful construction on C.

Example A.1.12. **Tuple** is a playful construction on **Set**.

Example A.1.13. $\overline{\mathbf{Play}}$ is a playful construction on $\mathbf{Dig}_*$

It is helpful to think of a playful construction on C as a kind of presheaf on $\mathbf{List}(C)^{\mathrm{op}}$. Accordingly, we will make use of some standard notation for presheaves. Given a playful construction F on C, and a sequence $a = (a_i)_{i\in[n]}$ of objects of C, we'll refer to the elements $x \in F(a)$ as sections of F over a, and say $a = \mathrm{dom}(x)$. Given a map $f\colon a \to a'$ in $\mathbf{List}(C)$ and an element x of F with domain a, we'll follow the usual convention of writing $x{\restriction}_f$ for $F(f)(x)$. Similarly, given a subset X of a', we'll write $f^{-1}X$ for $\{x \in a\colon x{\restriction}_f \in X\}$. If a' is a subsequence of a and f is the canonical map from a to a' then we'll also write $x{\restriction}_{a'}$ in place of $x{\restriction}_f$.

Example A.1.14. If x is a play in $A \otimes B \otimes C$, then $x{\restriction}_{AC}$ is the play in $A \otimes C$ consisting of those moves of x which are moves in A or C, taken in the same order in which they appear in x.

Remark A.1.15. *This notation follows a different convention to that introduced in definition 3.4.3; for example, what was denoted $x{\restriction}_{AC}$ in example*

A.1.14 here would be $x\restriction_{\{1,3\}}$ in that notation. To avoid conflict, we shall stick to the notation just introduced throughout this appendix. However, the reader should bear in mind that there is a difference of notation when comparing the results of this appendix with the remarks in section 3.4

Reformulating definition A.1.10 in this language gives

DEFINITION A.1.16. *Let F be a presheaf on $\mathbf{List}(C)^{\mathrm{op}}$, and let $(A_i \xrightarrow{f_i} A'_i)_{i\in[n]}$ be a sequence of maps in C.* F satisfies the shape condition *at $(f_i)_{i\in[n]}$ iff, for each section s of F over $(A'_i)_{i\in[n]}$ and each list $(x^i)_{i\in[n]}$ of sections of F over the A_i, with $x_i\restriction_{f_i} = s\restriction_{A'_i}$ for each i, there is a unique section x of F over $\bigotimes_{i\in[n]} A_i$ such that $x\restriction_{A_i} = x^i$ and $x\restriction_{(f_i)_{i\in[n]}} = s$.*

LEMMA A.1.17. *Let F be a presheaf on $\mathbf{List}(C)^{\mathrm{op}}$ F is a playful construction on C iff it satisfies the shape condition at each list $(f_i)_{i\in[n]}$.* □

The shape conditions are equivalent to the existence of patchings for particular sieves, which suggests a further reformulation.

DEFINITION A.1.18. *Let $\underline{f} = (A_i \xrightarrow{f_i} A'_i)_{i\in[n]}$ be a sequence of maps in C. The* shapely sieve *$S(\underline{f})$ consists of all arrows out of $(A_i)_{i\in[n]}$ which factor through either the map $(f_i)_{i\in[n]}$ or one of the maps $(A_i)_{i\in[n]} \to A_i$.*

The playful topology *on $\mathbf{List}(C)^{\mathrm{op}}$ is the Grothendieck topology J generated by all the shapely sieves $S(\underline{f})$ for $\underline{f}$ a list of maps in C.*

PROPOSITION A.1.19. *The sheaves for the playful topology are the playful constructions on C.* □

If C has a terminal object, then only pullbacks involving this terminal object need to be considered. Restricting A.1.17 to this case gives

LEMMA A.1.20. *Let C be a category with a terminal object 1, and let F be a playful construction on C. Let s be a section of F over $1^{\otimes n}$ and let $(x^i)_{i\in[n]}$*

be sections of F over objects A_i of C, with $x_i{\restriction_!}$ equal to the i^{th} restriction of s for each i. Then there is a unique section x of F over $\bigotimes_{i\in[n]} A_i$ such that $x{\restriction_{A_i}} = x^i$ and $x{\restriction_{!^n}} = s$.

□

By elementary properties of pullbacks, any functor $F\colon \mathbf{List}(C) \to \mathbf{Set}$ satisfying the condition in the conclusion of lemma A.1.20 must be a playful construction; so this condition is necessary and sufficient.

Lemma A.1.20 can be unwound for **Play** as follows: Let x be a play in a tensor product $\bigotimes_{i\in[n]} D_i$ of digraphs. Then x is not completely determined by the restrictions $x{\restriction_{D_i}}$. Intuitively, x is formed by some interleaving of the sequences $x{\restriction_{D_i}}$, and so x also encodes information about the *shape* of this interleaving. A sensible choice of representative for this shape is the finite sequence obtained from x by forgetting all information about the identity of the terms, and only remembering which games they were played in. This is the play $x{\restriction_{!^n}}$ in $1^{\otimes n}$, called the *shape* of x. Then the statement above says that the shape of x, together with the restrictions $x{\restriction_{D_i}}$, completely determines x.

As an example of the use of lemma A.1.20, let x in $A \otimes B$ and y in $A \otimes C$ be plays, with $x{\restriction_A} = y{\restriction_A}$. Then there must be some *interleaving* $z \in U(A \otimes B \otimes C)$ of x and y, with $z{\restriction_{AB}} = x$ and $z{\restriction_{AC}} = y$. For, by lemma A.1.20, it is enough to prove this in the case $A = B = C = 1$. This reduces the problem to showing that, given injective maps of ordered sets $I \to J$ and $I \to K$, there is a total order on the pushout which restricts to the orders on J and on K, and that is clear. Now using the symmetric monoidal structure on **Dig** it follows that for any sequences $A = (A_i)_{i\in[l]}$, subsets I and J of $[l]$ with $I \cup J = [l]$ and plays $x \in U((A_i)_{i\in I})$ and $y \in U((A_i)_{i\in J})$ such that $x{\restriction_{(A_i)_{i\in I\cap J}}} = y{\restriction_{(A_i)_{i\in I\cap J}}}$ there is some interleaving $z \in U(A)$ such that $z{\restriction_{(A_i)_{i\in I}}} = x$ and $z{\restriction_{(A_i)_{i\in J}}} = y$.

A.2 Glueing with sections

Given a playful construction, there is an associated **fc**-graph which encodes much of the combinatorics of that construction.

Definition A.2.1. *Let* $F\colon \mathbf{List}(C) \to \mathbf{Set}$ *be a playful construction. The* **fc***-graph* $\mathbf{Sec}(F)$ *of sections of* F *has*

- *objects and vertical 1-cells given by the objects and arrows of* C*.*
- *horizontal 1-cells from* a *to* a' *given by the sections of* F *over* aa'*.*
- *2-cells filling the rectangle*

$$\begin{array}{ccccccc} a_0 & \xrightarrow{m_1} & a_1 & \xrightarrow{m_2} & \cdots & \xrightarrow{m_n} & a_n \\ {\scriptstyle f}\downarrow & & & & & & \downarrow{\scriptstyle f'} \\ a & & & & \xrightarrow[m]{} & & a' \end{array}$$

  given by sections θ *of* F *over* $(a_i)_{i\in[0,n]}$ *such that* $\theta\restriction_{ff'} = m$ *and for each* $i \in [n]$ $\theta\restriction_{a_{i-1}a_i} = m_i$*.*

- *nullary 2-cells, that is, those filling rectangles such as*

$$\begin{array}{ccc} a_0 & = & a_0 \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f'} \\ a & \xrightarrow[m]{} & a' \end{array}$$

  given by sections θ *of* F *over* a_0 *such that* $\theta\restriction_f = m\restriction_a$ *and* $\theta\restriction_{f'} = m\restriction_{a'}$*.*

Example A.2.2. Applying this to the playful construction **Tuple** of example A.1.8, we get the **fc**-graph $\textsc{mat} = \mathbf{Sec}(\mathbf{Tuple})$ of matrix placeholders, with

- objects given by sets.
- vertical 1-cells given by functions.

- horizontal 1-cells from X to X' given by pairs (x, x') with $x \in X$ and $x' \in X'$.

- a unique 2-cell filling the rectangle

$$\begin{array}{ccccccc} X_0 & \xrightarrow{(x_0,x'_1)} & X_1 & \xrightarrow{(x_1,x'_2)} & \cdots & \xrightarrow{(x_{n-1},x'_n)} & X_n \\ {\scriptstyle f}\downarrow & & & & & & \downarrow{\scriptstyle f'} \\ X & & \xrightarrow[(x,x')]{} & & & & X' \end{array}$$

  if $x'_i = x_i$ for each $i \in [n-1]$, $f(x_0) = x$ and $f'(x'_n) = x'$, and no 2-cells filling this rectangle otherwise. Letting $x_n = x'_n$, we can identify this 2-cell with the sequence $(x_i)_{i\in[0,n]}$.

Example A.2.3. In a similar way, applying this to the playful construction $\overline{\mathbf{Play}}$ of example A.1.9, we get the **fc**-graph $\mathbf{Sec}(\overline{\mathbf{Play}})$ with

- objects given by digraphs.

- vertical 1-cells given by maps of digraphs.

- horizontal 1-cells from D to D' given by plays in $D \otimes D'$.

- 2-cells filling the rectangle

$$\begin{array}{ccccccc} D_0 & \xrightarrow{m_1} & D_1 & \xrightarrow{m_2} & \cdots & \xrightarrow{m_n} & D_n \\ {\scriptstyle f}\downarrow & & & & & & \downarrow{\scriptstyle f'} \\ D & & \xrightarrow[m]{} & & & & D' \end{array}$$

  given by plays p in $\bigotimes_{i\in[0,n]} D_i$ such that $p\restriction_{ff'} = m$ and $p\restriction_{D_{i-1}D_i} = m_i$ for each $i \in [n]$.

- nullary 2-cells filling the rectangle

$$\begin{array}{ccc} D_0 & = & D_0 \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f'} \\ D & \xrightarrow[m]{} & D' \end{array}$$

given by plays p in D_0 such that $p\restriction_f = m\restriction_D$ and $p\restriction_{f'} = m\restriction_{D'}$.

There is a sense in which MAT is terminal amongst structures produced this way.

DEFINITION A.2.4. *Let* $F\colon \mathbf{List}(C) \to \mathbf{Set}$ *be a playful construction. Then the projection* $\pi_F\colon \mathbf{Sec}(F) \to$ MAT *acts*

- *on objects and vertical 1-cells by* $F\restriction_C$
- *on horizontal 1-cells by* $(a \xrightarrow{m} a') \mapsto (F(a) \xrightarrow{(m\restriction_a, m\restriction_{a'})} F(a'))$.
- *on 2-cells by sending*

$$\begin{array}{ccccccc} a_0 & \xrightarrow{m_1} & a_1 & \xrightarrow{m_2} & \cdots & \xrightarrow{m_n} & a_n \\ {\scriptstyle f}\downarrow & & \Downarrow\theta & & & & \downarrow{\scriptstyle f'} \\ a & & \xrightarrow[m]{} & & & & a' \end{array}$$

to

$$\begin{array}{ccccccc} F(a_0)(f) & \xrightarrow{(m_1\restriction_{a_0}, m_1\restriction_{a_1})} & F(a_1) & \xrightarrow{(m_2\restriction_{a_1}, m_2\restriction_{a_2})} & \cdots & \xrightarrow{(m_n\restriction_{a_{n-1}}, m_n\restriction_{a_n})} & F(a_n) \\ {\scriptstyle F}\downarrow & & \Downarrow(\theta\restriction_{a_i})_{i\in[0,n]} & & & & \downarrow{\scriptstyle F(f')} \\ F(a) & & \xrightarrow[(m\restriction_a, m\restriction_{a'})]{} & & & & F(a')\,. \end{array}$$

We'll also want to make use of a slightly simpler construction than **Sec**.

DEFINITION A.2.5. *Let* $F\colon C \to \mathbf{Set}$ *be any functor. Then the playpen* $\mathbf{SSec}(F)$ *of* simplified sections *of* F *is given by the pullback of the cospan* $I_*C \xrightarrow{I_*F} I_*\mathbf{Set} \xleftarrow{\eta}$ MAT*, where* $V_* \dashv I_*$ *is the adjunction introduced at the start of section 3.2*

EXAMPLE A.2.6. The playpen $\mathbf{SSec}(\mathbf{Play})$ played a key role in section 3.4, where it went by the simpler name P.

Definition A.2.7. *Let* $F\colon \mathbf{List}(C) \to \mathbf{Set}$ *be a playful construction. The* simplification map $S_F\colon \mathbf{Sec}(F) \to \mathbf{SSec}(F\restriction_C)$ *is the map induced from the commutative square*

$$\begin{array}{ccc} \mathbf{Sec}(F) & \xrightarrow{\pi_F} & \mathrm{MAT} \\ {\scriptstyle\eta}\downarrow & & \downarrow{\scriptstyle\eta} \\ I_*C & \xrightarrow[I_*(F\restriction_C)]{} & I_*\mathbf{Set} \end{array}$$

Unlike these **fc**-graphs $\mathbf{SSec}(F)$, the **fc**-graphs $\mathbf{Sec}(F)$ don't naturally have the structure of playpens, so the constructions of section 3.2 can't be applied to them. However, there is a closely related construction in the context of an **fc**-graph A, a playful construction F on a category C, and a map of **fc**-graphs $l\colon A \to \mathbf{Sec}(F)$. We shall assume that such a context is fixed for the remainder of this section.

Definition A.2.8. *The* glued **fc**-*graph* $\mathbf{Gl}'(l)$ *of* l *has*

- *objects given by pairs* (a, f)*, where* a *is an object of* A *and* f *is a vertical 1-cell in* $\mathbf{Sec}(F)$ *(that is, a map in* C*) with target* $l(a)$*.*
- *vertical 1-cells from* (a, f) *to* (a', f') *given by pairs* (k, g)*, where* $k\colon a \to a'$ *and* g *are vertical 1-cells in* A *and* $\mathbf{Sec}(F)$ *respectively such that* $f' \cdot g = l(k) \cdot f$*.*
- *horizontal 1-cells from* (a, f) *to* (a', f') *given by pairs* (m, ϕ) *such that* $m\colon a \to a'$ *is a horizontal 1-cell in* A *and* ϕ *is a 2-cell in* $\mathbf{Sec}(F)$ *filling some square*

$$\begin{array}{ccc} s(f) & \xrightarrow{s(\phi)} & s(f') \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle f'} \\ l(a) & \xrightarrow[l(m)]{} & l(a')\,. \end{array}$$

- *2-cells filling the rectangle*

$$\begin{array}{ccccccc} (a_0, f_0) & \xrightarrow{(m_1,\phi_1)} & (a_1, f_1) & \xrightarrow{(m_2,\phi_2)} & \cdots & \xrightarrow{(m_n,\phi_n)} & (a_n, f_n) \\ {\scriptstyle (k,g)}\downarrow & & & & & & \downarrow{\scriptstyle (k',g')} \\ (a, f) & & & & \xrightarrow[(m,\phi)]{} & & (a', f') \end{array}$$

given by pairs (θ, ρ) *where* θ *is a 2-cell filling the rectangle*

$$\begin{array}{ccccccc} a_0 & \xrightarrow{m_1} & a_1 & \xrightarrow{m_2} & \cdots & \xrightarrow{m_n} & a_n \\ {\scriptstyle k}\downarrow & & & & & & \downarrow{\scriptstyle k'} \\ a & & & & \xrightarrow[m]{} & & a' \end{array}$$

in A*, and* ρ *is a 2-cell filling the rectangle*

$$\begin{array}{ccccccc} s(f_0) & \xrightarrow{s(\phi_1)} & s(f_1) & \xrightarrow{s(\phi_2)} & \cdots & \xrightarrow{s(\phi_n)} & s(f_n) \\ {\scriptstyle f\cdot g}\downarrow & & & & & & \downarrow{\scriptstyle f'\cdot g'} \\ l(a) & & & & \xrightarrow[l(m)]{} & & l(a') \end{array}$$

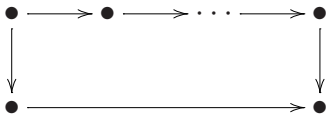
in $\mathbf{Sec}(F)$*, such that* $\rho\restriction_{f_0 f_1 \dots f_n} = l(\theta)$ *and* $\rho\restriction_{gg'} = \phi$*.*

There is no need for a separate clause for nullary 2-cells here: the final clause makes sense for nullary cells. We'll say that $\mathbf{Gl}'(l)$ *is* obtained by glueing along l*.*

Example A.2.9. Let 1 be the terminal **fc**-graph, and consider the map $l\colon 1 \to \textsc{mat}$, sending

- the object to $\{*\}$.
- the vertical 1-cell to the unique map $(\{*\} \to \{*\})$.
- the horizontal 1-cell to $(*, *)$.

- the 2-cell n filling the rectangle

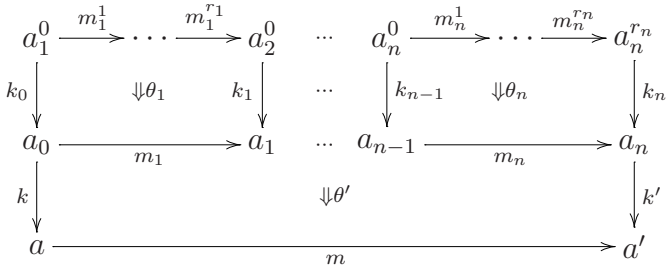


  to the constant tuple $(*)_{i\in[0,n]}$

Then $\mathbf{Gl}'(l) = \textsc{mat}$.

If A and l have good enough properties then this has a close relation to the gluing construction for playpens introduced in section 3.2.

Definition A.2.10. *Let A be a playpen and let $F\colon \mathbf{List}(C) \to \mathbf{Set}$ be a playful construction. A map $l\colon A \to \mathbf{Sec}(F)$ of* **fc***-graphs* preserves decomposition *if it restricts to a functor $V_*A \to C$ and for each decomposition*



of a cell θ in A, $l(\theta)\restriction_{(a_i^j)_{j\in[0,r_i]}} = l(\theta_i)$ and $l(\theta)\restriction_{(k_u(i))_{i\in[r]}} = l(\theta')\restriction_{(a_u(i))_{i\in[r]}}$ for any subsequence $(u(i))_{i\in[r]}$ of $[0,n]$ such that the sources of the maps $k_u(i)$ are all distinct.

Example A.2.11. The map l of example A.2.9 preserves decomposition.

Lemma A.2.12. *If a map l as in definition A.2.10 preserves decomposition then the map $S_F \cdot l$ is a map of playpens.*

Proof. S_F is the identity on vertical arrows, so $V_*(S_F \cdot l)$ is a functor. The remaining conditions imply that for each decomposition as in definition A.2.10 and each i and j, $l(\theta_i)\restriction_{a_i^j} = l(\theta)\restriction a_i^j$ and $l(\theta')\restriction_{a_i} = l(\theta)\restriction_{k_i}$, which is the required condition. □

Definition A.2.13. *Let l be a map in the context of definition A.2.10 which preserves composition. The comparison map $C(l)\colon \mathbf{Gl}'(l) \to \mathbf{Gl}(S_F \cdot l)$ acts by*

- *the identity on objects and vertical 1-cells*
- *sending the horizontal 1-cell (m, ϕ) to $(m, S_F(\phi))$*
- *sending the 2-cell (θ, ρ) to $(\theta, S_F(\rho))$.*

Theorem A.2.14. *The comparison map is an isomorphism.*

Proof. It is evidently an isomorphism on objects and vertical 1-cells, since it is the identity there. That it is an isomorphism on horizontal 1-cells and on 2-cells is an immediate consequence of lemma A.1.17. □

In particular, this shows that the **fc**-graphs $\mathbf{Gl}'(F)$ can in this context be given the structure of playpens. Also, if the playpen A has almost all composites (definition 4.2.3), then so does $\mathbf{Gl}'(l)$.

Example A.2.15. Let q_n be the map $R_n \to R_1^{\otimes n}$ sending moves in the i^{th} gap to moves in the i^{th} component and sending clockwise moves to blue moves and anticlockwise moves to red moves. Let $\textsc{alt}'\colon \textsc{ring} \to \mathbf{Sec}(\mathbf{Play})$ be the map of **fc**-graphs sending

- the unique object to R_1
- the unique vertical map to 1_{R_1}.
- the horizontal 1-cell p to $p\restriction_{q_2}$.
- the n-ary 2-cell c with $n > 0$ to $c\restriction_{q_{n+1}}$.

alt preserves composition, and $S_{\mathbf{Play}} \cdot \textsc{alt}' = \textsc{alt}$. Thus the **fc**-graph $\mathbf{Gl}(\textsc{alt})$ studied in section 3.4 is isomorphic to $\mathbf{Gl}'(\textsc{alt}')$.

Index

2-cell, **17**
  opcartesian, **94**
  weakly opcartesian, **93**, 101

Accessible digraph, *see* Digraph, accessible
Almost all composites, *see* Playpen, with almost all composites
Arbegla, **38**

Bicoloured digraph, *see* Digraph, bicoloured

Cartesian colax monoidal (or CCM) functor, **143**
Category
  of games, 109
Category of games, **5**, 54, 83, 90, 132
  of Conway games, 6, 136, **137**, 139
  of impartial games, **114**
Continual 1-cell, *see* Horizontal 1-cell, continual
Conway game, *see* Category of Conway games
Copycat
  play, *see* Play, copycat
  strategy, *see* Strategy, copycat

Decomposition, 8, **37**, 43
  for **fc**-multicategories, **65**
  for plain multicategories, **63**
  for playpens, 8, **68**
  preservation of, **155**
Digraph, **21**–**25**, 143
  accessible, **139**
  bicoloured, **85**, 90, 138
  indiscrete, **66**
  wellfounded, **130**
Discrete opfibration, 30, **61**
Doesn't get stuck, *see* Getting stuck
Double category
  strict, **95**
  virtual, *see* Multicategory, **fc**-weak, 8, **91**, 92

Enough composites, *see* Playpen, with enough composites
Exponentiability, **8**

fc-graph, *see* Graph, **fc**-
fc-multicategory, *see* Multicategory, **fc**-
fm-graph, *see* Graph, **fm**-
fm-multicategory, *see* Multicategory, plain

Game, *see* Category of games
Gap, **77**
Getting stuck, **130**
Gluing
  for categories, **69**
  for **fc**-graphs of sections, **153**
  for playpens, **69**
Graph
  fc-, **17**, 153
    of sections, **150**
  fm-, **15**
  T-, **14**

Horizontal 1-cell, **17**
  continual, **129**, 138
Horizontal monoid, **102**
  total, **106**

Impartial game, *see* Category of impartial games

Knot map, **83**

Locally ordered **fc**-multicategory, *see* Multicategory, **fc**, locally ordered

Multicategory, 12, **14**
  fc-, 7, **18**, 54
    locally ordered, **95**
  free, **56**
  plain, **16**
  unwirable, *see* Unwirability

Orderly play, *see* Play, orderly

Plain multicategory, *see* Multicategory, plain
Play, **29–35**
  copycat, **89**
  orderly, **86**, 137
Playful construction, **147**, 153
Playful topology, **148**
Playpen, 8, 54, 65, **66**
  of simplified sections, **152**
  with almost all composites, **97**
  with enough composites, **97**, 100
Positional game, 22, **122**
Pre-colax functor, **145**
Pre-composite, **93**, 128
Pre-unit, **93**
Promonoidal category, **25**, 64
  unbiased, **26**

Representability, **91**, 100

Restriction
along a gap map, **78**
along a map of digraphs, **88**, 147
Rib game, **122**, *see also* Category of rib games
Ring digraph, **76**, 117

Schedule, **135**
Section, *see* Graph, **fc**-, of sections
simplified, *see* Playpen, of simplified sections
Shape condition, **148**
Shapely sieve, **148**
Simplification map, **153**
Simulation, **88**, 117, 138
Slice
of **fc**-multicategories, **103**, 118
Strategy, 5, **126**, 132
copycat, 5, **102**, 127
winning, **137**
String game, **77**
Structuring **fm**-graph, **106**

Tensor product
of digraphs, **23**
of rib games, **123**
T-graph, *see* Graph, T-
Thread map, **79**
T-multicategory, *see* Multicategory
Total horizontal monoid, *see* Horizontal monoid, total
Tree, **34**, 134

Unwirability, **41**
of maps, **46**
of **fc**-multicategories, **65**
of multicategories, 51, **59**
of plain multicategories, **63**
of plain multicategories, 42, **64**
Unwiring, **37**
of a map, **42**
of a map of multicategories, **58**
weak, **57**

Vertical 1-cell, **17**
Virtual double category, *see* Multicategory, **fc**-

Weak double category, *see* Double category, weak
Weak unwiring, *see* Unwiring, weak
Weakly opcartesian 2-cell, *see* 2-cell, weakly opcartesian
Wellfounded digraph, *see* Digraph, wellfounded
Winning strategy, *see* Strategy, winning