POLARIZED CATEGORY THEORY, MODULES, AND GAME SEMANTICS

J.R.B. COCKETT AND R.A.G. SEELY

18 January 2007

ABSTRACT. Motivated by an analysis of Abramsky-Jagadeesan games, the paper considers a categorical semantics for a polarized notion of two-player games, a semantics which has close connections with the logic of (finite cartesian) sums and products, as well as with the multiplicative structure of linear logic. In each case, the structure is polarized, in the sense that it will be modelled by two categories, one for each of two polarities, with a module structure connecting them. These are studied in considerable detail, and a comparison is made with a different notion of polarization due to Olivier Laurent: there is an adjoint connection between the two notions.

Contents

Introduction

The idea of developing an algebraic and proof theoretic approach to game theory has a certain level of irony, since games, viewed as combinatorial structures, are regarded as being endowed with sufficient worldliness that they pass muster as a respectable semantics. The temptation to reinvent them as a type theory and thereby turn this notion of semantics on its head was irresistible.

The realization that the logic of (polarized) games is a just a subtle modification of the logic of products and sums [CS00] suggested to us that there was a rather different approach

Research partially supported by NSERC, Canada. Diagrams in this paper were produced with the help of the XY-pic macros of K. Rose and R. Moore, the *diagxy* macros of M. Barr, and TEXCAD by G. de Montmollin.

to understanding these games. Considering the logical complexities of the commuting conversions of the logic of products and coproducts, $\Sigma\Pi$ [CS00], and the connections of that system with games, it is not unnatural to ask if one can avoid the conversions by introducing some extra type constraints on the type theory, and if the resulting system has a game-theoretic interpretation. The answer is in fact more far-reaching even than the example of $\Sigma\Pi$ might lead one to expect. The key lay in providing a categorical semantics for these subtly changed sums and products. This meant that we had to understand the categorical meaning of polarization and the related notion of "focus".

Of course, once one puts the question in these terms, the answer inevitably is staring one in the face. The logic of a two player game cries out to be interpreted as a module between two categories. The problem then is to transport standard categorical notions into this "polarized" world. Central to this was the idea of an "inner adjoint" which has the universal properties of an adjoint but in a polarized sense.

It remained however to find a voice for this way of telling the story of polarized games amidst the altogether more practical uses of game theory and a community very focused (and rightly so) on applications of games. This paper has had a long period of gestation, and many of the ideas underlying the story we wished to tell were just beneath the surface in the community anyway, so it was not surprising that as we began to talk openly about our perspective on these games [C00, C02a, C02b], Olivier Laurent published his work on "polarized linear logic" [L02][1].

Laurent's view of polarization, while being very similar to ours, at the same time was also subtly different. His view of polarization was heavily influenced by Girard's view of and grouping of the connectives of linear logic. Consequently his work struck a familiar cord with many linear logicians. Furthermore, Laurent used a Hyland-Ong style game theoretic models to provide a semantics.

Inevitably, our view of the polarization of the connectives was rather different. We had taken as our starting point the games used by Abramsky and Jagadeesan [AJ92] and this had lead us to a rather different organization of the same basic material. At the end of the paper we explain the relationship between the two approaches. The main difference is two-fold: we emphasise different operators, and we include operators not included in Laurent's presentation. Almost all these operators may be seen in the simple finitary game model that serves as our motivation in the first section. This perspective makes some important aspects of these game models explicit, which were implicit in previous treatments, such as focalization and the subtly different notions of sums and products possible in the polarized setting. The latter can be interpreted as different communication strategies which we discuss in sections 4, 5.

For example, we describe a "depolarization" process which can take a polarized model and produce a $*$-autonomous category, *i.e.* a non-polarized model of (multiplicative) linear logic. The navigation of the polarized additive connectives and their role in depolarization is sufficiently complex that, without a careful treatment of these connectives, it is not easy to see what properties are required of the polarized model to ensure the depolarization has multiplicatives. The choices made, for example, by Laurent do not support depolarization with multiplicatives. On the other hand, the choices implicit in Abramsky and Jagadeesan

[1]Laurent has published sevaral variants of his polarized logic; for definiteness our comments refer to [L02].

are precisely sufficient to provide a depolarization with multiplicatives. However, they are not sufficient to deliver additives, which requires a different additive structure, as we shall discuss in section 8.2.

Furthermore, we note that our notion of polarization is compatible with the co-Kleisli construction in the presence of the "exponentials" **!** and **?** — in fact, given a polarized game category with a suitable notion of **!** and **?**, there is a polarized co-Kleisli construction which lifts the semantics to include these exponentials. Such a construction doesn't "type" in the Laurent setting, and cannot work in such a simple fashion.

Lest anyone think otherwise, we should make clear that we do not take the view that one approach is superior to the other. There may be many notions of polarization, each with its own virtues and special properties, and we hope adherence to one will not preclude readers from the delights of others. Laurent polarization provides a series of categorical doctrines which are parallel to ours. In fact, they are linked to our doctrines by adjunctions which use the family construction to freely add non-polarized additives.

The publication of Laurent's work did cause us to wonder again whether there was sufficient left in the story we wished to tell. Laurent's work had, for example, provided a very compact (one sided) sequent presentation for games. We had felt that our sequent presentation was a highlight — indeed a novelty — of our work. But although we can no longer claim originality for providing a sequent logic for these polarized games, we do claim our systems have some interesting features. One dubious distinction is that our systems have many more rules! However there are some good reasons for this. We take a very basic approach to these logics, making sure that they correspond transparently to their categorical semantics. However, this is not the real source of their size; rather, it is our continued insistence that these systems need have neither negation nor a commutative multiplicative structure. Thus the calculii we consider are more general those presented in Laurent's work; but more important is that ours are very modular (features are added only as needed). We think that the real gain is in the explicit nature of the resulting logic.

The story of this game theory has been told many times, often with the intent of getting the reader to the applications in the semantics of programming as fast as possible [H97, A97]. In this context it has become usual to regard games as being combinatorial structures and thus to be imbued with sufficient concreteness to be passable as a semantics. This is not the story we wish to tell here: we take (in common with Laurent) a very proof theoretic approach and when we talk of semantics we are thinking of the categorical models of the proof theory which have as little claim to concreteness as the proof theory itself. To be sure, we regard it as remarkable and fortuitous that the initial models have a concrete combinatorial description. However, our primary interest in them stems from the fact that they are the result of general constructions and that these constructions allow movement, at a general level, from one categorical doctrine to another.

We think that the view that polarized games have a natural categorical semantics in modules is original to our approach to this subject. Furthermore, the use of a new notion of adjunction ("inner adjoints") to characterize polarized sums and products, which we regard as a starting point for understanding the categorical semantics of polarized games, is also original to our particular way of telling the story.

The paper is structured as follows: in Part I, we set out the basic situation for polarized categories, starting with the basic game model. This structure captures the essence of the notions of polarity and of polarized sums and products. In fact, Part I is a self-contained entity, giving the key ideas of the paper. In Part II we extend this structure; in terms of the communication interpretation of games, we include communication along multiple input and output channels, arriving at the notion of a polarized polycategory. Representing this polystructure with appropriate tensors and pars is done in Part III on representability. Finally, in Part IV, we extend the theory, in particular describing depolarization, the polarized exponentials, and finally the connection with Olivier Laurent's approach to polarity. The reader familiar with his work ought to refer to Table 17 first, to get at least an idea of how our notions (and notation) compare with his.

Part I The basic game situation

1. Basic polarized games

To begin with we shall present a type system which we claim accounts for the basic structure of 2–player input–output games, of the sort studied (in the context of semantics of linear logic) by Abramsky and Jagadeesan [A97]. We consider this as an example of a general process; we shall probe this special case as an illustration, but do not regard it as exhausting the techniques or ideas behind this paper. For example, although we do not consider Conway games in this paper, these appear to be susceptible to a similar treatment albeit with rather different type theory.[2] We shall start with a simple type theory for games; however it is not sufficient to handle game constructors such as tensor and par. In later sections we shall show how those may be handled by a richer system, which may be more easily understood after the simpler system has been presented. In addition, we shall present a categorical semantics for these type theories in terms of polarized sums and products.

The games we wish to abstract have two players: O the "opponent" and P the "player", each of which has associated moves. When the morphisms between these games are viewed as processes, it is natural to think of the moves as messages which are being passed between processes. It is then usual to classify these messages from a "system centric" perspective: those which originate from the environment and those which are generated by the process or system itself. In the codomain of a morphism it is possible to identify the system messages with player moves and environment messages with the opponent moves. However, in the domain these roles are completely reversed: system message are identified with opponent moves and environment messages with player moves. An important characteristic of a game is whether the opponent or player starts, as this determines the direction of the initial message.

Since initially we shall not consider type constructors like "internal hom" we cannot follow the more usual approach of coding a morphism up as a strategy for a single game (of type $A \multimap B$). Instead we have to explicitly define the morphisms between our games. To facilitate this, in the next section we shall think of games as types and morphisms between games as proofs, derivations, or terms, in a manner familiar from type theory. The fact that there are opponent and player games necessitates that the type theory has opponent and player sequents which accommodate the different sorts of games which are available. In addition our basic type theory will have two constructions which allow us to build games as trees whose paths consist of alternating sequences of O-moves and P-moves. Given any finite family $\{X_i\}_{i\in I}$ of O-games, we can construct a P-game $\coprod_{i\in I} X_i$ and dually given any finite family $\{Y_j\}_{j\in J}$ of P-games, we can construct an O-game $\prod_{j\in J} Y_j$. To allow a connection between O-sequents and P-sequents we shall need "mixed" or "cross" sequents which operate between O-games and P-games.

To illustrate the structure we have in mind, we shall start with a variant[3] of a well-known

[2] At times we shall refer to "combinatorial games"; in this paper, by that phrase we shall mean Abramsky-Jagadeesan style games, not Conway games.

[3] We shall use the abbreviation "AJ games" to refer to our finitary variant, using "Abramsky-Jagadeesan games" for the infinitary games of [AJ92, A97], or for the minor variants that appear in other papers by one or both of those authors.

model, *viz.* Abramsky-Jagadeesan games [A97] which actually gives the initial categorical model of the basic type theory we shall introduce. We start by explaining this model from the present point of view to motivate the abstractions we shall make with our type theories. In case the reader gets the wrong impression, however, we should remark here that the infinitary games, with strategies (and winning strategies), may be presented as a model of the following framework as well, although we think the finitary variant gives a clearer model, and has the additional virtue of being a free model of the basic logic. The syntax we present below has no explicit reference to strategies, however. The appropriate level of categorical generalization of strategies is still unclear to us, and will have to await a sequel. An idea of a possible approach may be found in the "glueing" example, 3.0.2.

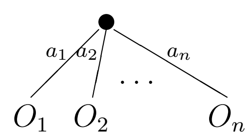
1.1. Polarized (finite) AJ games. Here a game may be regarded as a finite labeled bipartite tree: the nodes are partitioned into player states and opponent states and a labeled edge is required to start in a different partition from where it ends. When the root is a player node we shall call the game a player game and similarly if the root is an opponent node we shall call it a opponent game.

We shall use several notations for these games.

- A player game is denoted

$$P = \{a_i\colon O_i \mid i \in I\} = \bigsqcup_{i \in I} a_i\colon O_i$$

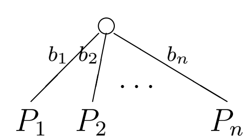
where each O_i is an opponent game. Moreover, supposing $I = \{1, 2, \ldots, n\}$, we could represent this by the following graph.



- An opponent game is denoted

$$O = (b_j\colon P_j \mid j \in J) = \bigsqcap_{j \in J} b_j\colon P_j$$

where each P_j is a player game. Again, this might be represented as this graph.



The binary versions of the basic operations are $O \sqcup O'$, which takes two opponent games and produces a player game, and $P \sqcap P'$, which takes two player games and produces an opponent game. The atomic games are given when the index sets are empty. We shall denote these by $\mathbf{0} = \bigsqcup_{\emptyset} = \{\ \}$ and $\mathbf{1} = \prod_{\emptyset} = (\ )$. Graphically, these are just leaves on a tree.

Given a game G there is a dual game $\overline{G}$ which is obtained by swapping products for sums and overlining the component indicators (where we assume that double overlining is the identity). Thus, we have:

$$\begin{array}{rclclcl} \overline{P} & = & \overline{\bigsqcup_{i\in I} a_i{:}\, O_i} & = & \prod_{i\in I} \overline{a_i}{:}\, \overline{O_i} & = & (\overline{a_i}{:}\, \overline{O_i} \mid i \in I) \\ \overline{Q} & = & \overline{\prod_{j\in J} b_j{:}\, P_j} & = & \bigsqcup_{j\in J} \overline{b_j}{:}\, \overline{P_j} & = & \{\overline{b_j}{:}\, \overline{P_j} \mid j \in J\} \end{array}$$

Our basic game type theory will abstract just this basic $\bigsqcup - \prod$ structure, but these games carry some additional structure which we will present in a later section, and which motivates the multiplicative extension of the basic type theory.

1.2. Maps and strategies. The usual way to specify maps between these games is *via* strategies and counter-strategies. However we shall adopt a somewhat different approach by directly describing the morphisms between games. Strategies can then be recovered as morphisms from the final game $\mathbf{1}$ (and counter-strategies as morphisms to the initial game $\mathbf{0}$): using the closed structure which we introduce later we can recover the usual definition of the morphisms (see Proposition 7.2.2).

[**Opponent maps:**]

$$\begin{pmatrix} b_1 & \mapsto & h_1 \\ & \cdots & \\ b_m & \mapsto & h_m \end{pmatrix} : O \longrightarrow (b_1{:}\, P_1, \ldots, b_m{:}\, P_m)$$

where each $h_i{:}\, O \longrightarrow P_i$ is a mixed map. We shall occasionally use the in-line notation $(b_i{:}\, h_i)_{i\in I}{:}\, O \longrightarrow (b_i{:}\, P_i \mid i \in I)$. Note that the displayed notation has the advantage of not needing subscripts, since the tokens may play that role themselves.

[**Mixed maps:**] These are either of the form

$$\overrightarrow{a_k} \cdot g{:}\, O \longrightarrow \{a_1{:}\, O_1, \ldots, a_n{:}\, O_n\}$$

where $k \in \{1, \ldots, n\}$, and $g{:}\, O \longrightarrow O_k$ is an opponent map, or

$$\overleftarrow{b_k} \cdot f{:}\, (b_1{:}\, P_1, \ldots, b_n{:}\, P_n) \longrightarrow P$$

where $k \in \{1, \ldots, n\}$, and $f{:}\, P_k \longrightarrow P$ is a player map. When the subscript is not necessary (being specified by the token itself) we may drop it.

[**Player maps:**]

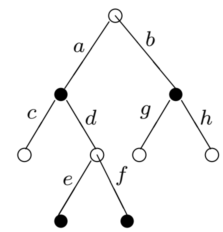
$$\begin{Bmatrix} a_1 & \mapsto & h_1 \\ & \cdots & \\ a_m & \mapsto & h_m \end{Bmatrix} : \{a_1{:}\, O_1, \ldots, a_m{:}\, O_m\} \longrightarrow P$$

where each $h_j{:}\, O_j \longrightarrow P$ is a mixed map. We shall occasionally use the in-line notation $\{a_j{:}\, h_j\}_{j\in J}{:}\, \{a_j{:}\, O_j \mid j \in J\} \longrightarrow P$.

1.2.1. Remark. Note that this notation gives in effect a notational comparison between CCS and the π-calculus on one side, and the categorical notions of product and coproduct on the other. This is intended, and reflects a basic intuition behind this work. This is made even more explicit in the thesis of Craig Pastro [P03].

1.2.2. Example. Here is a map between two opponent games:

$$\begin{pmatrix} a & \mapsto & \overrightarrow{c} \cdot () \\ b & \mapsto & \overrightarrow{g} \cdot () \end{pmatrix} : \circ \longrightarrow$$

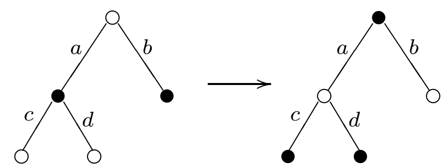


Here are four mixed maps between these given games:

1. $\overleftarrow{a} \cdot \begin{Bmatrix} c & \mapsto & \overrightarrow{b} \cdot () \\ d & \mapsto & \overrightarrow{b} \cdot () \end{Bmatrix}$ 2. $\overrightarrow{b} \cdot ()$

3. $\overrightarrow{a} \cdot \begin{pmatrix} c & \mapsto & \overleftarrow{b} \cdot \{\} \\ d & \mapsto & \overleftarrow{b} \cdot \{\} \end{pmatrix}$ 4. $\overleftarrow{b} \cdot \{\}$

:



1.3. Compositions. Next we define four compositions, *via* rewriting:

[**Opponent opponent composition**]

$$g \; ; \begin{pmatrix} b_1 & \mapsto & h_1 \\ & \cdots & \\ b_m & \mapsto & h_m \end{pmatrix} \Longrightarrow \begin{pmatrix} b_1 & \mapsto & g \; ; h_1 \\ & \cdots & \\ b_m & \mapsto & g \; ; h_m \end{pmatrix}$$

[**Opponent mixed composition**]

$$\begin{pmatrix} b_1 & \mapsto & h_1 \\ & \cdots & \\ b_m & \mapsto & h_m \end{pmatrix} ; \overleftarrow{b_k} \cdot f \Longrightarrow h_k \; ; f \quad \text{and} \quad g' \; ; \overrightarrow{a} \cdot g \Longrightarrow \overrightarrow{a} \cdot (g' \; ; g)$$

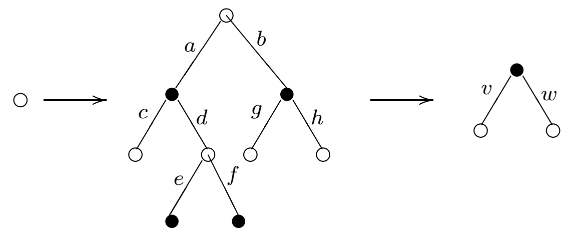
[**Mixed player composition**]

$$\overrightarrow{a_k} \cdot g \; ; \begin{Bmatrix} a_1 & \mapsto & h_1 \\ & \cdots & \\ a_m & \mapsto & h_m \end{Bmatrix} \Longrightarrow g \; ; h_k \quad \text{and} \quad \overleftarrow{b} \cdot f \; ; f' \Longrightarrow \overleftarrow{b} \cdot (f \; ; f')$$

[**Player player composition**]

$$\begin{Bmatrix} a_1 & \mapsto & h_1 \\ & \cdots & \\ a_m & \mapsto & h_m \end{Bmatrix} ; f \Longrightarrow \begin{Bmatrix} a_1 & \mapsto & h_1 \; ; f \\ & \cdots & \\ a_m & \mapsto & h_m \; ; f \end{Bmatrix}$$

1.3.1. Example. Here is a reduction of an opponent map composed with a mixed map.



$$
\begin{aligned}
\begin{pmatrix} a & \mapsto & \overrightarrow{c}\cdot() \\ b & \mapsto & \overrightarrow{g}\cdot() \end{pmatrix} ; \overleftarrow{b}\cdot \begin{pmatrix} g & \mapsto & \overrightarrow{w}\cdot() \\ h & \mapsto & \overrightarrow{v}\cdot() \end{pmatrix} &\Longrightarrow \overrightarrow{g}\cdot() ; \begin{pmatrix} g & \mapsto & \overrightarrow{w}\cdot() \\ h & \mapsto & \overrightarrow{v}\cdot() \end{pmatrix} \\
&\Longrightarrow () ; \overrightarrow{w}\cdot() \\
&\Longrightarrow \overrightarrow{w}\cdot(() ; ()) \\
&\Longrightarrow \overrightarrow{w}\cdot()
\end{aligned}
$$

It is an easy inductive argument to show that this is a confluent and terminating rewriting which eliminates the composition (as we shall shortly see this is a cut-elimination procedure). Furthermore, these rewritings satisfy the associative law in all the configurations which are possible. (See [CS00] for proofs for a similar system — in fact those proofs carry over to the present context, and even become simpler since the permuting conversions of [CS00] are absent in the present context.)

1.3.2. Lemma.

(i) *The above rewriting on maps terminates.*

(ii) *The above rewriting on maps is confluent.*

(iii) *The associative law is satisfied by all composible triples.*

To establish categorical structure for games and morphisms, we must exhibit the appropriate identity maps.

Given a player object $P = \{a_i\colon O_i \mid i \in I\}$ we define the identity map $1_P = \{\overrightarrow{a_i}\cdot 1_{O_i}\}_{i\in I}$; given $O = (b_i\colon P_i \mid i \in I)$ we define its identity map $1_O = (\overleftarrow{b_i}\cdot 1_{P_i})_{i\in I}$. We then have:

1.3.3. Lemma. *In any possible composition with an identity, the identity acts as a neutral element with respect to that composition.*

As will be seen in section 3, this means that we have two categories, the player and the opponent category, linked by a module (see Definition 3.0.1). In that section we give a complete characterization of the categorical models which in addition to being a module must possess polarized products and sums.

2. Basic polarized game logic

Before we look at categorical structures, we shall approach this *via* type theory, presenting the logic as a sequent calculus over a type theory. In effect, we are presenting the graph structure,

Table 1: Basic polarized additives

$$\frac{}{A \vdash_p A}\ \textit{atomic identities}\ \frac{}{A \vdash_o A}$$

$$\frac{\left\{X_i \underset{o\ p}{\mapsto} Y\right\}_{i\in I}}{\bigsqcup_{i\in I} X_i \vdash_p Y}\ \textit{cotuple} \qquad \frac{\left\{X \underset{o\ p}{\mapsto} Y_i\right\}_{i\in I}}{X \vdash_o \bigsqcap_{i\in I} Y_i}\ \textit{tuple}$$

$$\frac{X \vdash_o Y_k}{X \underset{o\ p}{\mapsto} \bigsqcup_{i\in I} Y_i}\ \textit{injection} \qquad \frac{X_k \vdash_p Y}{\bigsqcap_{i\in I} X_i \underset{o\ p}{\mapsto} Y}\ \textit{projection}$$

where $k \in I$, $I \neq \emptyset$

together with a cut elimination process, which will motivate and justify the categorical structure presented later. This basic game logic will be a bit peculiar since we shall need three kinds of sequents:

Player sequents: These take the form:

$$X \vdash_p Y$$

where X and Y are player propositions.

Opponent sequents: These are dual to the player sequents, they take the form:

$$V \vdash_o W$$

where V and W are opponent propositions.

Cross sequents: These are self-dual and have the form:

$$V \underset{o\ p}{\mapsto} Y$$

where V is an opponent proposition and Y is a player proposition.

The valid inferences are generated from the rules in Table 1, which are a "graded" version of $\Sigma\Pi$ [CS00].

Notice that the rules are symmetric: the symmetry is given by swapping the direction of the sequents while at the same time swapping "player" for "opponent" and $\bigsqcup$ for $\bigsqcap$. This symmetry arises from an underlying categorical duality.

The logic has four cut rules (Table 2) which correspond to those permitted by the types. The first two arise as cuts respectively in the player and opponent sequents. The last two are the two possible cuts on the cross sequent. These correspond categorically to the compositions (or rather the left and right actions) expected of a module.

Table 2: Basic cut rules

$$\frac{X \vdash_p Y \quad Y \vdash_p Z}{X \vdash_p Z}\ p\text{-}cut \qquad \frac{X \vdash_o Y \quad Y \vdash_o Z}{X \vdash_o Z}\ o\text{-}cut$$

$$\frac{X \underset{o\ p}{\longmapsto} Y \quad Y \vdash_p Z}{X \underset{o\ p}{\longmapsto} Z}\ cp\text{-}cut \qquad \frac{X \vdash_o Y \quad Y \underset{o\ p}{\longmapsto} Z}{X \underset{o\ p}{\longmapsto} Z}\ oc\text{-}cut$$

Table 3: Basic terms

$$\frac{}{1_A\!:: A \vdash_p A}\quad atomic \qquad identities \quad \frac{}{1_A\!:: A \vdash_o A}$$

$$\frac{\left\{h_i\!:: X_i \underset{o\ p}{\longmapsto} Y\right\}_{i\in I}}{\{a_i\colon h_i\}_{i\in I}\!:: \bigsqcup_{i\in I} a_i\colon X_i \vdash_p Y}\ cotuple \qquad \frac{\left\{h_i\!:: X \underset{o\ p}{\longmapsto} Y_i\right\}_{i\in I}}{(b_i\colon h_i)_{i\in I}\!:: X \vdash_o \prod_{i\in I} b_i\colon Y_i}\ tuple$$

$$\frac{g\!:: X \vdash_o Y_k}{\overrightarrow{a_k} \cdot g\!:: X \underset{o\ p}{\longmapsto} \bigsqcup_{i\in I} a_i\colon Y_i}\ injection \qquad \frac{f\!:: X_k \vdash_p Y}{\overleftarrow{b_k} \cdot f\!:: \prod_{i\in I} b_i\colon X_i \underset{o\ p}{\longmapsto} Y}\ projection$$

where $k \in I$, $I \neq \emptyset$

$$\frac{f\!:: X \vdash Y \quad g\!:: Y \vdash Z}{f\ ;g\!:: X \vdash Z}\ cut$$

where $\vdash$ represents the appropriate type of sequent for each of the four cut rules

2.1. A term logic. In fact more is true. As for $\Sigma\Pi$ we may assign terms to this logic (Tables 3 and 4): however, where $\Sigma\Pi$ needed commuting conversions this logic does not because the type system makes the conversions impossible. Essentially this means that it is possible to have combinatorial models for these game processes as there are no manipulations once cut has been eliminated. The terms and term rewrites are similar to the ones we listed for polarized AJ games. To reduce the overload strain on colons, we use :: to denote the term-type membership relation, so $t\!:: U \vdash V$ will mean that t is a term of type $U \vdash V$, where U (say) may be of the form $a\colon X$. Then we can assert that cut elimination steps preserve the equivalence on terms induced by these rewrites.

We shall see some examples of these rewrites (in a more general context, and using proof circuits) in sections 5, 5.3 and 6.5; as they are in principle similar to those in [CS00], we shall leave further examples and details to the reader. However, it is now a simple exercise to prove the following.

Table 4: Basic rewrites

$$
\begin{array}{rcl}
f \,;\, 1 & \Longrightarrow & f \\
1 \,;\, f & \Longrightarrow & f \\
g' \,;\, \overrightarrow{a} \cdot g & \Longrightarrow & \overrightarrow{a} \cdot (g' \,;\, g) \\
\overleftarrow{b} \cdot f \,;\, f' & \Longrightarrow & \overleftarrow{b} \cdot (f \,;\, f') \\
\{h_i\}_{i\in I} \,;\, f & \Longrightarrow & \{h_i \,;\, f\}_{i\in I} \\
g \,;\, (h_i)_{i\in I} & \Longrightarrow & (g \,;\, h_i)_{i\in I} \\
\overrightarrow{a_k} \cdot g \,;\, \{h_i\}_{i\in I} & \Longrightarrow & g \,;\, h_k \\
(h_i)_{i\in I} \,;\, \overleftarrow{b_k} \cdot f & \Longrightarrow & h_k \,;\, f
\end{array}
$$

2.1.1. Theorem. *The basic game logic satisfies cut elimination, and furthermore, the cut elimination process satisfies the Church-Rosser property.*

Anticipating the definitions of section 3, we can see that a categorical model for this logic must consist of two categories, the player category $\mathbf{X}_p$ and the opponent category $\mathbf{X}_o$, and a "module" $\widehat{\mathbf{X}}\colon \mathbf{X}_o \longrightarrow \mathbf{X}_p$. (Such a "module" behaves much as one would expect from the ring theory notion of a 2-sided module, but we shall soon make the notion more concrete, in section 3.) Furthermore, for each index set I we have functors $\bigsqcup_I\colon \mathbf{X}_o^I \longrightarrow \mathbf{X}_p$ and $\bigsqcap_I\colon \mathbf{X}_p^I \longrightarrow \mathbf{X}_o$ with the following natural correspondences:

$$
\frac{\{X_i \longrightarrow\!\!\!\!\!\!\!\!+\;\;\; Y\}_{i\in I} \;\text{ in } \widehat{\mathbf{X}}}{\overline{\bigsqcup_{i\in I} X_i \longrightarrow Y \quad \text{in } \mathbf{X}_p}}
\qquad\qquad
\frac{\{X \longrightarrow\!\!\!\!\!\!\!\!+\;\;\; Y_i\}_{i\in I} \;\text{ in } \widehat{\mathbf{X}}}{\overline{X \longrightarrow \bigsqcap_{i\in I} Y_i \quad \text{in } \mathbf{X}_o}}
$$

These correspondences have the following consequences. Let $\bigsqcup_1 B$ and $\bigsqcap_1 A$ be respectively the unary game sum and game product. Then there are bijections:

$$
\begin{array}{c}
\bigsqcup_1 Y \longrightarrow X \quad \text{in } \mathbf{X}_p \\
\hline\hline
Y \longrightarrow\!\!\!\!\!\!\!\!+\;\;\; X \quad \text{in } \widehat{\mathbf{X}} \\
\hline\hline
Y \longrightarrow \bigsqcap_1 X \quad \text{in } \mathbf{X}_o
\end{array}
$$

which shows that $\bigsqcup_1$ is left adjoint to $\bigsqcap_1$ and that the module is generated by these functors. Notice also that we have the correspondence:

$$
\begin{array}{c}
\{\bigsqcup_1 Y_i \longrightarrow X\}_{i\in I} \\
\hline\hline
\{Y_i \longrightarrow\!\!\!\!\!\!\!\!+\;\;\; X\}_{i\in I} \\
\hline\hline
\bigsqcup_{i\in I} Y_i \longrightarrow X
\end{array}
$$

which shows that $\mathbf{X}_p$ has I-indexed coproducts of objects of the form $\bigsqcup_1 B$. This gives:

2.1.2. Proposition. *A model for the basic game logic is equivalent to an adjunction* $\bigsqcup_1 \dashv \prod_1 : \mathbf{X}_o \longrightarrow \mathbf{X}_p$ *in which (I-indexed) coproducts of objects of the form* $\bigsqcup_1 Y$ *exist in* $\mathbf{X}_p$ *and (I-indexed) products of objects of the form* $\prod_1 A$ *exist in* $\mathbf{X}_o$.

This means there are plenty of models since any adjunction between a category $\mathbf{X}_o$ with coproducts and $\mathbf{X}_p$ with products will automatically produce a model. Clearly any category with products and coproducts will be a model (using the identity adjunction).

The proof theory, of course, allows us to generate free models from arbitrary modules. The initial model (generated from the module between empty categories) is the polarized finite AJ games described earlier. This may be proved directly, but will be left here as an exercise, since we shall prove it later by another route when we refine our view of these games and can link it to the original approach to the subject used in [AJ92] which uses strategies and counter-strategies.

We shall abstract the notion of model outlined above to develop the theory of polarized categories, and more specifically of polarized game categories, which is the correct domain for considering the semantics for (our sort of) polarized games.

2.2. Remark. One might wonder (as we did) whether a useful type theory may be based on cross sequents of the opposite type: $X \underset{p\ o}{\vdash\!\!\!\rightarrow} Y$. There are philosophical reasons for rejecting these (as there may well be reasons for wanting them), but from the present point of view, we shall merely point out that such a type theory blocks the inductive construction of identity derivations, such as $X \sqcup Y \vdash_p X \sqcup Y$, and generally will have an unsatisfactory categorical semantics (consider that coproducts cannot have injections due to typing conflicts, for example).

3. Polarized categories

To arrive at a semantic doctrine for these basic polarized games, we shall need some of the theory of "polarized categories". Although not our primary motivation, it also seems that this is a possible doctrine within which to develop a semantics for Girard's original notion of ludics [G01].

3.0.1. Definition. *A polarized category* $\mathbf{X} = \langle \mathbf{X}_o, \mathbf{X}_p, \widehat{\mathbf{X}} \rangle$ *consists of a pair of categories* $\mathbf{X}_o$, $\mathbf{X}_p$ *together with a module* $\widehat{\mathbf{X}} : \mathbf{X}_o \longrightarrow \mathbf{X}_p$.

A module $\widehat{\mathbf{X}} : \mathbf{X}_o \longrightarrow \mathbf{X}_p$ is a profunctor $\widehat{\mathbf{X}} : \mathbf{X}_o \longrightarrow \mathbf{X}_p$, that is to say, a functor $\mathbf{X}_o^{\mathrm{op}} \times \mathbf{X}_p \longrightarrow \mathrm{Sets}$. We can regard such a module as a span $\mathrm{Obj}(\mathbf{X}_o) \longleftarrow \widehat{\mathbf{X}} \longrightarrow \mathrm{Obj}(\mathbf{X}_p)$ in the category Sets, subject to the usual module closure condition: this may be regarded as a set of (formal) arrows whose domain is an object of $\mathbf{X}_o$ and whose codomain is an object of $\mathbf{X}_p$. These arrows must be closed under precomposition with arrows of $\mathbf{X}_o$ and under postcomposition with arrows of $\mathbf{X}_p$, and must satisfy the evident associativity and identity equations. We shall write module arrows with a small vertical hatch on the shaft of the arrow: $A \longrightarrow\!\!\!\!\!\!\!+\;\; B$. Given a polarized category $\langle \mathbf{X}_o, \mathbf{X}_p, \widehat{\mathbf{X}} \rangle$, there is an obvious dual polarized category $\langle \mathbf{X}_o, \mathbf{X}_p, \widehat{\mathbf{X}} \rangle^{\mathrm{op}} = \langle \mathbf{X}_p^{\mathrm{op}}, \mathbf{X}_o^{\mathrm{op}}, \widehat{\mathbf{X}}^{\mathrm{op}} \rangle$, where the dual $\widehat{\mathbf{X}}^{\mathrm{op}}$ is the same formal set of arrows as $\widehat{\mathbf{X}}$, but now regarded as having the opposite direction: $B \longrightarrow\!\!\!\!\!\!\!+\;\; A$.

3.0.2. Example. The following example (and several sequels throughout the paper) may be particularly of interest to readers familiar with double glueing. Suppose $\mathbf{C}$ is a category, with distinguished objects I and J, and a distinguished set $\mathcal{K}$ of morphisms $\mathsf{I} \longrightarrow \mathsf{J}$. We shall define a polarized category $\mathbf{X} = \mathcal{G}(\mathbf{C}, \mathcal{K})$ as follows. An object of $\mathbf{X}_o$ is a pair $(\mathcal{R}, X)$, for X an object of $\mathbf{C}$ and $\mathcal{R} \subseteq \mathbf{C}(\mathsf{I}, X)$. A morphism $(\mathcal{R}, X) \xrightarrow{f} (\mathcal{R}', X')$ is given by $X \xrightarrow{f} X'$ in $\mathbf{C}$ so that $r \in \mathcal{R} \Rightarrow r \,;\, f \in \mathcal{R}'$. Dually, an object of $\mathbf{X}_p$ is a pair $(Y, \mathcal{S})$, Y an object of $\mathbf{C}$, $\mathcal{S} \subseteq \mathbf{C}(Y, \mathsf{J})$; a morphism $(Y, \mathcal{S}) \xrightarrow{g} (Y'\mathcal{S}')$ is given by $Y \xrightarrow{g} Y'$ so that $s' \in \mathcal{S}' \Rightarrow g \,;\, s' \in \mathcal{S}$. Finally, a module morphism $(\mathcal{R}, X) \overset{h}{-\!\!\!+\!\!\!\rightarrow} (Y, \mathcal{S})$ is given by $X \xrightarrow{h} Y$ in $\mathbf{C}$ so that for all $r \in \mathcal{R}, s \in \mathcal{S}$, $r \,;\, f \,;\, s \in \mathcal{K}$. It is easy to show that this is indeed a module.

This example may be given using a slightly different language. For an object X of $\mathbf{C}$, for morphisms $f \in \mathbf{C}(\mathsf{I}, X)$, $g \in \mathbf{C}(X, \mathsf{J})$, say that f is "orthogonal" to g, $f \perp_X g$, if $f \,;\, g \in \mathcal{K}$. Clearly, for any $X \xrightarrow{h} Y$, $f \perp_X h \,;\, g$ if and only if $f \,;\, h \perp_Y g$. Such a notion of orthogonality is equivalent to the specification of a distinguished set $\mathcal{K}$; to get $\mathcal{K}$ from $\perp$, just set $\mathcal{K} = \{f \,;\, g \mid f \perp g\}$. Then the definition of module maps becomes f so that $r \,;\, f \perp_Y s$ for all r, s, equivalently so that $r \perp_X f \,;\, s$ for all r, s.

Anticipating section 3.2, note that there are two constructions taking us between $\mathbf{X}_o$ and $\mathbf{X}_p$: $(\mathcal{R}, X)^* = (X, \mathcal{R}^*)$, where $\mathcal{R}^* = \{h \colon X \longrightarrow \mathsf{J} \mid r \perp_X h,\ \forall r \in \mathcal{R}\}$ and $(Y, \mathcal{S})_* = (\mathcal{S}_*, Y)$, where $\mathcal{S}_* = \{k \colon \mathsf{I} \longrightarrow Y \mid k \perp_Y s,\ \forall s \in \mathcal{S}\}$. It is easy to show the following natural bijections, establishing that these are adjoint, and moreover, they characterize the module structure.

$$\frac{\dfrac{(\mathcal{R}, X)^* \xrightarrow{h} (Y, \mathcal{S})}{(\mathcal{R}, X) \overset{h}{-\!\!\!+\!\!\!\rightarrow} (Y, \mathcal{S})}}{(\mathcal{R}, X) \xrightarrow{h} (Y, \mathcal{S})_*}$$

3.0.3. Definition. *A polarized functor* $F = \langle F_o, F_p, \widehat{F} \rangle \colon \langle \mathbf{X}_o, \mathbf{X}_p, \widehat{\mathbf{X}} \rangle \longrightarrow \langle \mathbf{X}'_o, \mathbf{X}'_p, \widehat{\mathbf{X}}' \rangle$ *consists of two functors* $F_o \colon \mathbf{X}_o \longrightarrow \mathbf{X}'_o$, $F_p \colon \mathbf{X}_p \longrightarrow \mathbf{X}'_p$, *and a module morphism* $\widehat{F} \colon \widehat{\mathbf{X}} -\!\!\!+\!\!\!\rightarrow \widehat{\mathbf{X}}'$, viz.

$$\widehat{F} \colon x \overset{m}{-\!\!\!+\!\!\!\rightarrow} y \mapsto F_o(x) \overset{\widehat{F}(m)}{-\!\!\!-\!\!\!+\!\!\!-\!\!\!\rightarrow} F_p(y)$$

satisfying $F_o(a) \,;\, \widehat{F}(m) \,;\, F_p(b) = \widehat{F}(a \,;\, m \,;\, b)$ *for* $x' \xrightarrow{a} x$ *in* $\mathbf{X}_o$ *and* $y \xrightarrow{b} y'$ *in* $\mathbf{X}_p$.

3.0.4. Definition. *A polarized natural transformation* $\alpha \colon \langle F_o, F_p, \widehat{F} \rangle \longrightarrow \langle F'_o, F'_p, \widehat{F}' \rangle$ *consists of a pair* $\alpha := \langle \alpha_o, \alpha_p \rangle$ *of natural transformations* $\alpha_o \colon F_o \longrightarrow F'_o$, $\alpha_p \colon F_p \longrightarrow F'_p$ *making the following commute for any module arrow* $m \colon A -\!\!\!+\!\!\!\rightarrow B$.

$$\begin{tikzcd}
F_o(A) \arrow[r, "\widehat{F}(m)", "\shortmid"{marking}] \arrow[d, "\alpha_o(A)"'] & F_p(B) \arrow[d, "\alpha_p(B)"] \\
F'_o(A) \arrow[r, "\widehat{F}'(m)"', "\shortmid"{marking}] & F'_p(B)
\end{tikzcd}$$

The collection of polarized categories, functors, and natural transformations forms a 2-category which we shall call **PolCat**. Note that this 2-category of polarized categories is (equivalent to) the slice category **Cat**/2, where 2 is the 2-point lattice regarded as a category.

3.0.5. Remark. Although we have a 2-category **PolCat**, it will not be the case that all notions appropriate for the polarized setting will be the usual notions interpreted in **PolCat**. In the next section, we shall see a central example of this phenomenon: polarized sums and products are not the usual notions interpreted in **PolCat**, but will require a new universal property. Later, in Section 4.2, we shall see another important instance of this, when we come to interpret the notions of polarized polycategories and polarized modules — again, the appropriate notions are not merely interpretations in an appropriate 2-category. Since the "pure" category theory is somewhat "skewed" by the polarized notions, keeping the games interpretation in mind is an excellent guide.

3.1. Inner and outer adjoints; polarized products and sums. In considering polarized structure, it turns out that a mixed notion (partially polarized, partially not) is of use. Consider how we ought to add polarized products and sums (especially with the example of AJ games in mind).

3.1.1. Definition. *A polarized category $\mathbf{X} = \langle \mathbf{X}_o, \mathbf{X}_p, \widehat{\mathbf{X}} \rangle$ is said to have I-indexed polarized products (for a set I) if there is a functor $\prod_I : \mathbf{X}_p^I \longrightarrow \mathbf{X}_o$ (also denoted $\prod_{i \in I}$) with the following natural correspondences:*

$$\frac{\left\{ X \overset{f_i}{\longrightarrow\!\!\!\!\!\!\!\!+\;\;} Y_i \right\}_{i \in I} \qquad \text{in } \widehat{\mathbf{X}}}{X \xrightarrow{(f_i)_I} \prod_I Y_i \qquad \text{in } \mathbf{X}_o}$$

$\mathbf{X}$ *is said to have I-indexed polarized sums if the dual polarized category $\mathbf{X}^{\mathrm{op}}$ has I-indexed polarized products.* $\mathbf{X}$ *is said to have all finite polarized sums and products if it has I-indexed sums and products for all finite sets I.*

Note that in this definition, the polarized sums and products are *selected*, rather than given by a universal property alone. However, we shall see that they do satisfy an appropriate universal property, once we have the right notion of adjunction to describe this situation.

Although we shall not need extensions of this definition, it is obvious that we can define arbitrary (not necessarily finite) polarized sums and products, and indeed, polarized limits and colimits for more general diagrams, in a similar fashion.

One fact that strikes one immediately about this notion is that it is *not* polarized in the most natural sense: *viz.* this is not the natural notion of limit in the 2-category **PolCat**. To say a polarized category $\mathbf{X}$ has polarized products and sums is not to require that the diagonal (polarized) functor have an adjoint in **PolCat**, but rather that it has a pair of "mixed" functors ("mixed" in the sense that they switch polarity) each of which sets up the expected bijection. This notion may be abstracted as follows.

3.1.2. Definition. *Suppose $F : \mathbf{X} \longrightarrow \mathbf{Y}$ is a polarized functor, and that $G := \langle G_o, G_p \rangle$ is a pair of functors*

$$G_o : \mathbf{Y}_p \longrightarrow \mathbf{X}_o \qquad G_p : \mathbf{Y}_o \longrightarrow \mathbf{X}_p$$

(note that G is not polarized). Then we say F has an inner adjoint G*, or equivalently that G has an* outer adjoint F*, if there are natural bijections*

$$
\begin{array}{lr}
F_o(X) \relbar\joinrel\mapstochar\joinrel\rightarrow Y' & \text{in } \widehat{\mathbf{Y}} \\
\hline\hline
X \longrightarrow G_o(Y') & \text{in } \mathbf{X}_o
\end{array}
\qquad\qquad
\begin{array}{lr}
Y \relbar\joinrel\mapstochar\joinrel\rightarrow F_p(X') & \text{in } \widehat{\mathbf{Y}} \\
\hline\hline
G_p(Y) \longrightarrow X' & \text{in } \mathbf{X}_p
\end{array}
$$

It is now a simple matter to verify that $\mathbf{X}$ has I-indexed polarized products and sums if $\Delta_I\colon \mathbf{X} \longrightarrow \mathbf{X}^I$ has an inner adjoint. It is worth noting that this notion of inner–outer adjunction does not compose.

Inner adjoints do have a universal property:

3.1.3. Proposition. *To say that a polarized functor $F\colon \mathbf{X} \longrightarrow \mathbf{Y}$ has an inner adjoint is precisely to say that there are object functions $G_o\colon \mathbf{Y}_p \longrightarrow \mathbf{X}_o$, $G_p\colon \mathbf{Y}_o \longrightarrow \mathbf{X}_p$, and natural families of module maps $\epsilon_{Y'}\colon F_oG_o(Y') \relbar\joinrel\mapstochar\joinrel\rightarrow Y'$, $\eta_Y\colon Y \relbar\joinrel\mapstochar\joinrel\rightarrow F_pG_p(Y)$, for all Y in $\mathbf{Y}_o$, Y' in $\mathbf{Y}_p$, so that for any module maps $g\colon F_o(X) \relbar\joinrel\mapstochar\joinrel\rightarrow Y'$, $f\colon Y \relbar\joinrel\mapstochar\joinrel\rightarrow F_p(X')$, there are unique maps $g^\flat\colon X \longrightarrow G_o(Y')$ in $\mathbf{X}_o$, $f^\sharp\colon G_p(Y) \longrightarrow X'$ in $\mathbf{X}_p$, making the following diagrams commute.*

$$
\begin{tikzcd}
F_o(X) \arrow[d, "F_o(g^\flat)"'] \arrow[dr, "g", "|" marking] & \\
F_oG_o(Y') \arrow[r, "\epsilon_{Y'}"', "|" marking] & Y'
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
Y \arrow[r, "\eta_Y", "|" marking] \arrow[dr, "f"', "|" marking] & F_pG_p(Y) \arrow[d, "F_p(f^\sharp)"] \\
& F_p(X')
\end{tikzcd}
$$

We can express this in a different manner. Suppose $F\colon \mathbf{X} \longrightarrow \mathbf{Y}$ is an ordinary functor; we can define modules $F^*\colon \mathbf{X} \longrightarrow \mathbf{Y}$ and $F_*\colon \mathbf{Y} \longrightarrow \mathbf{X}$ as follows: $X \relbar\joinrel\mapstochar\joinrel\rightarrow Y$ in F^* is a triplet $\langle X, F(X) \longrightarrow Y, Y\rangle$ and $Y \relbar\joinrel\mapstochar\joinrel\rightarrow X$ in F_* is a triplet $\langle Y, Y \longrightarrow F(X), X\rangle$. Then if F is a polarized functor, maps $F_o(X) \relbar\joinrel\mapstochar\joinrel\rightarrow Y'$ (for X in $\mathbf{X}_o$ and Y' in $\mathbf{Y}_p$) are maps of the composite module ${F_o}^* \otimes \widehat{\mathbf{Y}}$. Likewise maps $Y \relbar\joinrel\mapstochar\joinrel\rightarrow F_p(X')$ (for Y in $\mathbf{Y}_o$, X' in $\mathbf{X}_p$) are maps of the composite module $\widehat{\mathbf{Y}} \otimes {F_p}_*$. With this language, we can state the defining property of an inner adjoint as follows.

3.1.4. Proposition. *A polarized functor $F\colon \mathbf{X} \longrightarrow \mathbf{Y}$ has an inner adjoint if and only if there are module equivalences ${F_o}^* \otimes \widehat{\mathbf{Y}} \cong G_{o*}$ and $\widehat{\mathbf{Y}} \otimes {F_p}_* \cong {G_p}^*$, for some functors $G_o\colon \mathbf{Y}_p \longrightarrow \mathbf{X}_o$ and $G_p\colon \mathbf{Y}_o \longrightarrow \mathbf{X}_p$.*

We are now in a position to state the obvious corollary that inner (and outer) adjoints are unique up to unique isomorphisms, as with ordinary adjoints.

3.1.5. Corollary. *Suppose a polarized functor $F\colon \mathbf{X} \longrightarrow \mathbf{Y}$ has inner adjoints given by $(G_o, G_p, (\ )^\sharp, (\ )^\flat)$ and $(G'_o, G'_p, (\ )^{\sharp'}, (\ )^{\flat'})$. Then $G_o \cong G'_o$ and $G_p \cong G'_p$ are natural equivalences satisfying the obvious coherence conditions. On objects, these equivalences are given by unique isomorphisms.*

The proof is straightforward, and is left to the reader.

3.2. Modules given by adjunction. We can now return to consider polarized coproducts and products. First, note that as these are given by an inner adjoint, they are unique up to a unique isomorphism. Unary polarized coproducts and products play a special role.

First, consider the identity (polarized) functor on a polarized category $1_{\mathbf{X}}: \mathbf{X} \longrightarrow \mathbf{X}$; to say that it has an inner adjoint is precisely to say that the module $\widehat{\mathbf{X}}$ is given by an (ordinary) adjunction. For suppose $1_{\mathbf{X}}$ has an inner adjoint, given by $(\ )_*: \mathbf{X}_p \longrightarrow \mathbf{X}_o$ and $(\ )^*: \mathbf{X}_o \longrightarrow \mathbf{X}_p$. Then we have the following natural bijections.

$$\begin{array}{cc}
Q \longrightarrow P_* & \text{in } \mathbf{X}_o \\
\hline\hline
Q \mathrel{-\!\!\!+\!\!\!\rightarrow} P & \text{in } \widehat{\mathbf{X}} \\
\hline\hline
Q^* \longrightarrow P & \text{in } \mathbf{X}_p
\end{array}$$

The converse is obvious. Furthermore, it is clear that this adjunction is given by $\prod_1: \mathbf{X}_p \longrightarrow \mathbf{X}_o$ and $\coprod_1: \mathbf{X}_o \longrightarrow \mathbf{X}_p$, where 1 is a singleton set. So $(\ )^* = \coprod_1$ and $(\ )_* = \prod_1$ are "switch polarity" functors, *i.e.* inner adjoint to the identity. (One is tempted to call these functors "Pierre" and "Gaston", for if we think of polarized categories as describing games, these correspond to moves of the sort "apres vous, Gaston".) Then the adjunction

$$\begin{tikzcd}
\mathbf{X}_o \arrow[r, bend right, "\sqcup_1"'] \arrow[r, phantom, "\top"] & \mathbf{X}_p \arrow[l, bend right, "\sqcap_1"']
\end{tikzcd}$$

generates the module structure; it also shows the connection between polarized and non-polarized sums and products.

3.2.1. Lemma. *A polarized category has finite (I-indexed) polarized sums and products if and only if there is an adjunction $(\ )^* \dashv (\ )_*: \mathbf{X}_o \longrightarrow \mathbf{X}_p$ in which (I-indexed) coproducts of objects of the form Q^* exist in $\mathbf{X}_p$ and (I-indexed) products of objects of the form P_* exist in $\mathbf{X}_o$.*

As we saw with basic game types, the bijections

$$\begin{array}{cc}
\{\coprod_1 Q_i \longrightarrow P\}_{i \in I} & \text{in } \mathbf{X}_p \\
\hline\hline
\{Q_i \mathrel{-\!\!\!+\!\!\!\rightarrow} P\}_{i \in I} & \text{in } \widehat{\mathbf{X}} \\
\hline\hline
\coprod_{i \in I} Q_i \longrightarrow P & \text{in } \mathbf{X}_p
\end{array}$$

show that we have ordinary (non-polarized) sums (and products) of objects given by singleton (polarized) sums (and products).

Notice that this is the polarized categorical restatement of Proposition 2.1.2. In particular it allows us to conclude that a polarized category with polarized products and coproducts is precisely a model for our basic game logic.

3.2.2. Definition. *We shall call a polarized category which is generated by an adjoint in this fashion an* inner polarized category*.*

This means an inner polarized category has an inner adjoint to its identity function. For inner polarized categories we may construct polarized products and coproducts from ordinary products and coproducts:

3.2.3. Lemma. *An inner polarized category which has products in $\mathbf{X}_o$ and coproducts in $\mathbf{X}_p$ has polarized products and coproducts. The polarized products are constructed as $\prod_I P_i := \bigwedge_I P_{i*}$, where $\bigwedge$ is (ordinary) product in $\mathbf{X}_o$, and dually for polarized sums.*

3.2.4. Example. We already know the polarized category $\mathcal{G}(\mathbf{C},\mathcal{K})$ of Example 3.0.2 is generated by an adjunction; in addition it has polarized sums (respectively products) if $\mathbf{C}$ has ordinary sums (respectively products). Let

$$\bigsqcup_{i\in I}(\mathcal{R}_i, X_i) = (\textstyle\sum_i X_i, \{h\colon \sum_i X_i \longrightarrow \mathsf{J} \;\mid\; r_i \,;\, b_i \perp h \;\forall r_i \in \mathcal{R}_i\})$$

Then

$$\begin{array}{c} \{(\mathcal{R}_i, X_i) \xrightarrow{f_i} (Y,\mathcal{S})\}_{i\in I} \\ \hline\hline \bigsqcup_{i\in I}(\mathcal{R}_i, X_i) \xrightarrow{\{f_i\}_i} (Y,\mathcal{S}) \end{array}$$

In addition, $\mathcal{G}(\mathbf{C},\mathcal{K})_o$ also has ordinary sums and products if $\mathcal{C}$ does.

$$\sum_{i\in I}(\mathcal{R}_i, X_i) = (\textstyle\bigcup_i(\mathcal{R}_i \,;\, b_i), \sum_i X_i)$$

$$\prod_{i\in I}(\mathcal{R}_i, X_i) = (\langle \mathcal{R}_1, \ldots, \mathcal{R}_n\rangle, \textstyle\prod_i X_i)$$

Polarized products are handled dually, and $\mathcal{G}(\mathbf{C},\mathcal{K})_p$ has ordinary sums and products defined dually. If $\mathbf{C}$ is distributive, so are $\mathcal{G}(\mathbf{C},\mathcal{K})_o$ and $\mathcal{G}(\mathbf{C},\mathcal{K})_p$.

3.3. The 2-category of polarized games. We now wish to briefly consider the 2-category of polarized categories with finite polarized products and coproducts. As we think of an object in this category as a model for the basic polarized game logic we shall call the 2-category **PolGam** and refer to the objects as polarized game categories. We start by describing the functors of this 2-category:

3.3.1. Definition. *Suppose $\mathbf{X}, \mathbf{X}'$ are polarized categories with polarized sums. A polarized functor $F\colon \mathbf{X} \longrightarrow \mathbf{X}'$ preserves polarized sums if F_p preserves $\bigsqcup$ and $\widehat{F}$ preserves cotupling and injections. Explicitly, $F_p(\bigsqcup_I A_i) \cong \bigsqcup_I F_o(A_i)$ and the following diagrams commute, for $f_i\colon A_i \longrightarrow B$ $(i \in I)$, and $f\colon A \longrightarrow A_k$ $(k \in I)$.*

$$\begin{tikzcd}
 & \bigsqcup_I F_o(A_i) \arrow[dr, "\left\langle \widehat{F}(f_i)\right\rangle_I"] \arrow[dd, "\cong"] & \\
F_o(A) \arrow[ur, "b_k(F_o(f))"] \arrow[dr, "\widehat{F}(b_k(f))"'] & & F_p(B) \\
 & F_p(\bigsqcup_I A_i) \arrow[ur, "F_p(\langle f_i\rangle_I)"'] &
\end{tikzcd}$$

F preserving polarized products is defined dually.

Then polarized categories with finite polarized sums and products, polarized functors that preserve polarized sums and products, and polarized natural transformations form the 2-category **PolGam**.

3.3.2. PROPOSITION. *There is a forgetful 2-functor* $\mathbf{PolGam} \xrightarrow{\mathsf{U}} \mathbf{PolCat}$ *which has a left 2-adjoint*

$$\begin{tikzcd} \mathbf{PolCat} \arrow[r, bend right, "\mathsf{Gam}"'] \arrow[r, phantom, "\top"] & \mathbf{PolGam} \arrow[l, bend right, "\mathsf{U}"'] \end{tikzcd}$$

which constructs the free polarized game category generated by a polarized category.

PROOF. We shall sketch the construction of $\mathsf{Gam}(\mathbf{X})$. $\mathsf{Gam}(\mathbf{X})_o, \mathsf{Gam}(\mathbf{X})_p$ and $\widehat{\mathsf{Gam}(\mathbf{X})}$ are defined inductively (this is essentially just the construction of the free basic game types and terms generated by $\mathbf{X}$):

$$\begin{array}{rcl}
\mathrm{Ob}(\mathsf{Gam}(\mathbf{X})_o) & = & \mathrm{Ob}(\mathbf{X}_o) \cup \{\prod_I P_i \mid P_i \in \mathsf{Gam}(\mathbf{X})_p, i \in I, I \text{ a finite set}\} \\
\mathrm{Ob}(\mathsf{Gam}(\mathbf{X})_p) & = & \mathrm{Ob}(\mathbf{X}_p) \cup \{\coprod_I Q_i \mid Q_i \in \mathsf{Gam}(\mathbf{X})_o, i \in I, I \text{ a finite set}\} \\
\mathrm{Ar}(\mathsf{Gam}(\mathbf{X})_o) & = & \mathrm{Ar}(\mathbf{X}_o) \cup \{(f_i)_I : Q \longrightarrow \prod_I P_i \mid f_i : Q \longrightarrow\!\!\!\!\!+\ P_i \in \widehat{\mathsf{Gam}(\mathbf{X})}, i \in I, I \text{ a finite set}\} \\
\mathrm{Ar}(\mathsf{Gam}(\mathbf{X})_p) & = & \mathrm{Ar}(\mathbf{X}_p) \cup \{\langle f_i \rangle_I : \coprod_I Q_i \longrightarrow P \mid f_i : Q_i \longrightarrow\!\!\!\!\!+\ P \in \widehat{\mathsf{Gam}(\mathbf{X})}, i \in I, I \text{ a finite set}\} \\
\widehat{\mathsf{Gam}(\mathbf{X})} & = & \widehat{\mathbf{X}} \cup \{b_k(f) : Q \longrightarrow\!\!\!\!\!+\ \coprod_I Q_i \mid f : Q \longrightarrow Q_k \in \mathrm{Ar}(\mathsf{Gam}(\mathbf{X})_o), k \in I, I \text{ a finite set}\} \\
& & \cup\ \{p_k(f) : \prod_I P_i \longrightarrow\!\!\!\!\!+\ P \mid f : P_k \longrightarrow P \in \mathrm{Ar}(\mathsf{Gam}(\mathbf{X})_p), k \in I, I \text{ a finite set}\}
\end{array}$$

where we take the arrows mod the equivalence relation generated by the eight conversions of the basic game type theory. From this description of Gam, the unit η of the adjunction is clear and canonical (it is the evident inclusion). Given any polarized functor $F : \mathbf{X} \longrightarrow \mathsf{U}(\mathbf{X}')$ we construct $F^\sharp : \mathsf{Gam}(\mathbf{X}) \longrightarrow \mathbf{X}'$, a polarized functor that preserves polarized sums and products, defined inductively by sending the constructed $\coprod$ or $\prod$ in $\mathsf{Gam}(\mathbf{X})$ to the selected polarized sum or product in $\mathbf{X}'$. Likewise, given a polarized natural transformation $\alpha : F \longrightarrow F'$, we may construct a polarized natural transformation $\alpha^\sharp : F^\sharp \longrightarrow F'^\sharp$ in the same way. It is straightforward to show that this is indeed a 2-adjunction. ∎

It is interesting to note that one effect of the game construction is to produce a module which is generated by an adjoint. Indeed if the module has no cross maps then one side-effect of the construction is therefore to produce an adjunction between the two categories. In fact, if we restrict the construction to unary polarized products and coproducts the effect is to produce a "walking adjunction" [SS86]. And so this gives a game theoretic view of an old construction.

3.4. SOFTNESS. In [J95] Joyal describes a property "softness" which characterizes the structure of limits and colimits in free bicompletions of categories. A simplified version of this was presented in [CS00], dealing with the free finite product and sum completion. A simple variant of this property also applies to the polarized context.

3.4.1. DEFINITION. *A polarized game category* $\mathbf{X}$ *is soft if we have the following coproduct (in* Sets*).*

$$\widehat{\mathbf{X}}(\textstyle\prod_I X_i, \coprod_J Y_j) \cong \displaystyle\sum_{i \in I} \mathbf{X}_p(X_i, \textstyle\coprod_J Y_j) + \displaystyle\sum_{j \in J} \mathbf{X}_o(\textstyle\prod_I X_i, Y_j)$$

It is informative to compare this definition with that in [CS00], keeping in mind that certain configurations are ruled out by the typing; it will then be noticed that a pushout in the cartesian case must be replaced by a coproduct in the polarized case.

3.4.2. Definition. *Given a polarized game category* $\mathbf{X}$*, an opponent object* $A \in \mathbf{X}_o$ *is atomic if*

$$\widehat{\mathbf{X}}(A, \textstyle\bigsqcup_J X_j) \cong \displaystyle\sum_{j \in J} \mathbf{X}_o(A, X_j) \quad \textit{and} \quad \mathbf{X}_o(\textstyle\prod_K X'_k, A) \cong \emptyset$$

and dually a player object $B \in \mathbf{X}_p$ *is atomic if*

$$\widehat{\mathbf{X}}(\textstyle\prod_I X_i, B) \cong \displaystyle\sum_{i \in I} \mathbf{X}_p(X_i, A) \quad \textit{and} \quad \mathbf{X}_p(B, \textstyle\bigsqcup_K X'_k) \cong \emptyset$$

3.4.3. Definition. *A polarized category* $\mathbf{X}$ *is Whitman if every object of* $\mathbf{X}$ *is isomorphic to a game (*i.e. *to a* $\bigsqcup$ *or a* $\prod$*) of atomic objects, and if* $\mathbf{X}$ *is soft.*

Then one may characterize the image of the (faithful, though not full) 2-functor Gam in **PolGam** as follows.

3.4.4. Theorem. *A polarized game category* $\mathbf{X}$ *is isomorphic to* $\mathsf{Gam}(\mathbf{Y})$ *for a polarized category* $\mathbf{Y}$ *if and only if* $\mathbf{X}$ *is Whitman.*

Proof. The unit of the adjunction maps a polarized category into the game category constructed from it. It is easy to see from the construction that the atoms of the game category are exactly the objects of the polarized category and that this game category is soft (by construction).

Conversely, given a game category which is Whitman we claim it is equivalent to the game category on its atoms. The proof of this follows the steps in [CS00], which is to say it is a structural induction on the types using the softness to show that the maps from polarized products and to polarized coproducts are the same as those from the free types. ∎

Part II Multiple channels

4. The logic of polarized cut and its semantics

The simple game logic presented so far does not permit a process (a morphism or proof) to communicate along multiple input and output channels. Without this ability this game logic will be rather inexpressive. In this section we discuss how to add channels to the basic game logic. In order to do this the basic logic has to support different kinds of contexts within which a process can listen and send. We shall introduce a (possibly non-commutative) extension to the basic type theory, and we shall show that it is modeled by the AJ combinatorial games. It is worth noting that the exchange rule for the tensors and pars that we shall introduce may be added. Although it is usual for game models to be viewed as commutative, we regard this is an unnecessary restriction.

That this is a non-trivial logic follows immediately from the fact that it is modeled by MALL (multiplicative linear logic with additives). However, the point of the logic is that it affords more separation than MALL so that the categorical coherence problems are much simpler. This is the result of making polarity an explicit part of the system, as we have already seen with $\Sigma\Pi$-logic and the basic game logic. In particular coherence for the proof theory (that is the underlying free categories) for game types is a good deal simpler than for the additives in linear logic precisely because all the commuting conversions due to the additives have been removed by the type constraints. We shall discuss this after completing the description of the logic.

4.1. The logic of polarized cuts. In our extended game logic there are, as before, three types of sequent; however, this time the sequents have *contexts*, the forms of which need some preliminary explanation.

Player sequents: These take the form:

$$\Gamma \,/\, X \,\backslash\, \Gamma' \vdash_p \Delta$$

where Γ, Γ' are O-*phrases*, that is lists (or possibly bags) of opponent propositions, X is a player proposition, and Δ is a P-*phrase*, that is a list (or possibly a bag) of player propositions.

Opponent sequents: These are dual to the player sequents, they take the form:

$$\Gamma \vdash_o \Delta \,/\, Y \,\backslash\, \Delta'$$

where Γ is an O-phrase, Y is a opponent proposition, and Δ, Δ' are P-phrases.

Cross sequents: These are self-dual and have the form:

$$\Gamma \underset{o\ \ p}{\models\!\!\!\rightarrow} \Delta$$

where Γ is an O-phrase and Δ is a P-phrase.

The point to notice here is that in the P and O sequents, we allow a "context" of the opposite type, and since we wish our logic to allow for non-commutative operators (tensor and par), we allow that context to "surround" the active formula. If we were to assume commutative tensor and par, that would not be necessary, and "one-sided" contexts would suffice. This would also reduce the multiplicity of rules below, since we would no longer have to distinguish so many left-right cases. In all sequents, the left side is (primarily) O material, the right is (primarily) P material, with the proviso that "pure" O sequents allow an additional O proposition on the right, "pure" P sequents allow an additional P proposition on the left. These may be regarded as "in focus", or "active". A mixed or cross sequent has no active proposition.

In the following subsections, we describe the inference rules for this game logic. We shall use upper case letters at the beginning of the alphabet to denote atomic propositions, upper case at end of the alphabet to denote arbitrary propositions, and Greek upper case to denote lists of propositions. We proceed in several steps: the first indicates the basic context rules, including the cut rules, followed by the categorical (or rather polycategorical) semantics for cut. Then we give the rules for the basic ("polarized additive") game constructors. The remaining constructors, first the multiplicatives, then negation, and finally the exponentials, will follow in the next section on representing structure.

The logic has twenty four cut rules — although this may seem a lot, there is a simple underlying principle, which is we permit all possible well-typed, planar variants of the cut rule. As a single scheme, this would look something like this:

$$\frac{\Gamma \vdash_x \Delta X \Delta' \quad \Phi X \Phi' \vdash_y \Psi}{\Phi \Gamma \Phi' \vdash_z \Delta \Psi \Delta'}$$

where one of Δ, Φ must be empty, and one of Δ', Φ' must be empty (this is the planarity condition). There are only six choices of the types of entailment $\vdash_x, \vdash_y, \vdash_z$ that are permitted by the typing, and with four alternatives for each, we end up with the twenty four variants. These are illustrated in Table 5, where we leave to the reader the task of implementing the planarity condition. The *opp-cut*s are given in two versions, each with two variants, because the cut is being made into one side of the context or the other—a similar division is made for the dual *opo-cut*s.

Notice that all these rules preserve the basic duality of this logic obtained by swapping the direction of the sequents, exchanging player for opponent and products for sums. Furthermore, we have represented this duality in the left column right column symmetry in the table. Recall also that the exchange rule can be assumed, in which case the phrases are to be regarded as bags of propositions.

4.2. Polarized polycategories. Corresponding to the logic of polarized cut above is its categorical proof theory which is the notion of a polarized polycategory. In addition polarized polycategories, like polycategories, have a term logic which consists of polarized circuits. The purpose of this section is to introduce these ideas.

A polarized polycategory $\mathbf{X}$ consists of polycategories $\mathbf{X}_o$ and $\mathbf{X}_p$ as well as a polymodule $\widehat{\mathbf{X}}$. Each polyarrow in $\mathbf{X}_o$ is of the form

$$\Gamma \longrightarrow_o \Delta \,/\, Y \,\backslash\, \Delta'$$

Table 5: General cut rules

$$\frac{\Gamma_1 \,/\, P \backslash \Gamma_2 \vdash_p \Delta, X, \Delta' \quad \Phi \,/\, X \backslash \Phi' \vdash_p \Psi}{\Phi, \Gamma_1 \,/\, P \backslash \Gamma_2, \Phi' \vdash_p \Delta, \Psi, \Delta'}\ \textit{p-cut} \qquad \frac{\Gamma \vdash_o \Delta \,/\, X \backslash \Delta' \quad \Phi, X, \Phi' \vdash_o \Psi_1 \,/\, Q \backslash \Psi_2}{\Phi, \Gamma, \Phi' \vdash_o \Delta\Psi_1 \,/\, Q \backslash \Psi', \Delta'}\ \textit{o-cut}$$

$$\frac{\Gamma \vdash_o \Delta \,/\, X \backslash \quad \Phi, X, \Phi_1 \,/\, P \backslash \Phi_2 \vdash_p \Psi}{\Phi, \Gamma, \Phi_1 \,/\, P \backslash \Phi_2 \vdash_p \Delta, \Psi}\ \textit{lopp-cut} \qquad \frac{\Gamma \vdash_o \Delta_1 \,/\, Q \backslash \Delta_2, X, \Delta' \quad \,/\, X \backslash \Phi' \vdash_p \Psi}{\Gamma, \Phi' \vdash_o \Delta_1 \,/\, Q \backslash \Delta_2, \Psi, \Delta'}\ \textit{ropo-cut}$$

$$\frac{\Gamma \vdash_o \ \,/\, X \backslash \Delta' \quad \Phi_1 \,/\, P \backslash \Phi_2, X, \Phi' \vdash_p \Psi}{\Phi_1 \,/\, P \backslash \Phi_2, \Gamma, \Phi' \vdash_p \Psi, \Delta'}\ \textit{ropp-cut} \qquad \frac{\Gamma \vdash_o \Delta, X, \Delta_1 \,/\, Q \backslash \Delta_2 \quad \Phi \,/\, X \backslash \ \vdash_p \Psi}{\Phi, \Gamma \vdash_o \Delta, \Psi, \Delta_1 \,/\, Q \backslash \Delta_2}\ \textit{lopo-cut}$$

$$\frac{\Gamma \underset{o\,p}{\mapsto} \Delta, X, \Delta' \quad \Phi \,/\, X \backslash \Phi' \vdash_p \Psi}{\Phi, \Gamma, \Phi' \underset{o\,p}{\mapsto} \Delta, \Psi, \Delta'}\ \textit{cpc-cut} \qquad \frac{\Gamma \vdash_o \Delta \,/\, X \backslash \Delta' \quad \Phi, X, \Phi' \underset{o\,p}{\mapsto} \Psi}{\Phi, \Gamma, \Phi' \underset{o\,p}{\mapsto} \Delta, \Psi, \Delta'}\ \textit{occ-cut}$$

where in each rule where they appear, one of Δ, Φ is empty, and one of Δ', Φ' is empty.

having a sequence of objects Γ from $\mathbf{X}_o$ as its domain, and a sequence of objects all but one of which are from $\mathbf{X}_p$ as its codomain, with in addition one identified ("active", or "in focus") object from $\mathbf{X}_o$: Δ, Δ' from $\mathbf{X}_p$, Y from $\mathbf{X}_o$. This collection of arrows must contain an "identity" arrow $Y \longrightarrow_o \ \,/\, Y \backslash$ for each object Y of $\mathbf{X}_o$.

Dually, each polyarrow in $\mathbf{X}_p$ is of the form

$$\Gamma \,/\, X \backslash \Gamma' \longrightarrow_p \Delta$$

having a sequence of objects all but one of which are from $\mathbf{X}_o$ as its domain, with in addition one identified ("active", or "in focus") object from $\mathbf{X}_p$ in the domain, and a sequence of objects from $\mathbf{X}_p$ as its codomain: Γ, Γ' from $\mathbf{X}_o$, X, Δ from $\mathbf{X}_p$. This collection of arrows must contain an "identity" arrow $\,/\, X \backslash \ \longrightarrow_p X$ for each object X of $\mathbf{X}_p$.

We shall usually omit the subscripts on arrows in $\mathbf{X}_o, \mathbf{X}_p$, when the context makes them unnecessary.

Each polyarrow in the polymodule has the form

$$\Gamma \longrightarrow\!\!\!\!\!\!\mid\ \ \Delta$$

having a sequence Γ of objects from $\mathbf{X}_o$ in the domain and a sequence Δ of objects from $\mathbf{X}_p$ in the codomain.

These arrows may be composed in twenty-four ways, essentially as given by the twenty-four cut rules of game logic above. This may seem rather intimidating, but in essence the idea is quite simple: each of $\mathbf{X}_o$ and $\mathbf{X}_p$ allow composition much as ordinary polycategories do, but given the non-commutative nature of these sequents, there are minor variants caused by the placement of the active object in the sequents. In addition, $\mathbf{X}_o$ acts on $\widehat{\mathbf{X}}$ on the left, and $\mathbf{X}_p$ acts on $\widehat{\mathbf{X}}$ on the right, in evident ways. This amounts to all the well-typed planar variants of the following "generic" composition, allowing for the various types of arrows.

$$\frac{\Gamma \xrightarrow{f} \Delta X \Delta' \quad \Phi X \Phi' \xrightarrow{g} \Psi}{\Phi\Gamma\Phi' \xrightarrow{f\,;\,g} \Delta\Psi\Delta'}$$

where one of Δ, Φ is empty, and one of Δ', Φ' is empty. (This condition is referred to as the "planarity condition".) In terms of circuits, this is even simpler; it is the usual circuit cut (just join two wires which bear the same label), with the understanding now that the joined wires are of the same type (player or opponent, "solid" or "dotted").

There are "standard" unit and associativity conditions, analogous to those for ordinary polycategories. For simplicity, we illustrate these rules with "generic" versions. In these, we suppress the notation for which type of arrows are involved, where the composition (or cut) takes place, and so which type of composition or cut is involved; the reader is supposed to imagine all possible "well-typed" versions of these rules.

Recall that our compositions or cuts are supposed to be planar; we represent that by the convention that in these rules, an expression "$\Delta \mid \Gamma$" is to be understood as the trivial concatenation of a sequence and an empty sequence, the assumption being that one of Δ, Γ is empty.

(1)[idL]
$$\Gamma_1 \xrightarrow{f} \Gamma_2, A, \Gamma_3 \quad = \quad \dfrac{\Gamma_1 \xrightarrow{f} \Gamma_2, A, \Gamma_3 \qquad A \xrightarrow{i_A} A}{\Gamma_1 \xrightarrow{f\,;\,i_A} \Gamma_2, A, \Gamma_3}$$

(2)[idR]
$$\Gamma_1, A, \Gamma_2 \xrightarrow{f} \Gamma_3 \quad = \quad \dfrac{A \xrightarrow{i_A} A \qquad \Gamma_1, A, \Gamma_2 \xrightarrow{f} \Gamma_3}{\Gamma_1, A, \Gamma_2 \xrightarrow{i_A\,;\,f} \Gamma_3}$$

(3)[assoc]
$$\dfrac{\dfrac{\Gamma_1 \xrightarrow{f} \Gamma_2, A, \Gamma_3 \qquad \Delta_1, A, \Delta_2 \xrightarrow{g} \Delta_3, B, \Delta_4}{\Delta_1, \Gamma_1, \Delta_2 \xrightarrow{f\,;\,g} \Gamma_2, \Delta_3, B, \Delta_4, \Gamma_3} \qquad \Phi_1, B, \Phi_2 \xrightarrow{h} \Phi_3}{\Phi_1, \Delta_1, \Gamma_1, \Delta_2, \Phi_2 \xrightarrow{(f\,;\,g)\,;\,h} \Gamma_2, \Delta_3, \Phi_3, \Delta_4, \Gamma_3}$$

$$= \quad \dfrac{\Gamma_1 \xrightarrow{f} \Gamma_2, A, \Gamma_3 \qquad \dfrac{\Delta_1, A, \Delta_2 \xrightarrow{g} \Delta_3, B, \Delta_4 \qquad \Phi_1, B, \Phi_2 \xrightarrow{h} \Phi_3}{\Phi_1, \Delta_1, A, \Delta_2, \Phi_2 \xrightarrow{g\,;\,h} \Delta_3, \Phi_3, \Delta_4}}{\Phi_1, \Delta_1, \Gamma_1, \Delta_2, \Phi_2 \xrightarrow{f\,;\,(g\,;\,h)} \Gamma_2, \Delta_3, \Phi_3, \Delta_4, \Gamma_3}$$

(4)[inter1]
$$\dfrac{\Delta_1 \xrightarrow{g} \Delta_2, B, \Delta_3 \qquad \dfrac{\Gamma_1 \xrightarrow{f} \Gamma_2, A, \Gamma_3 \qquad \Phi_1, A, \Phi_2, B, \Phi_3 \xrightarrow{h} \Phi_4}{\Phi_1, \Gamma_1, \Phi_2, B, \Phi_3 \xrightarrow{f\,;\,h} \Gamma_2, \Phi_4, \Gamma_3}}{\Phi_1, \Gamma_1, \Phi_2, \Delta_1, \Phi_3 \xrightarrow{g\,;\,(f\,;\,h)} \Delta_2 \mid \Gamma_2, \Phi_4, \Delta_3 \mid \Gamma_3}$$

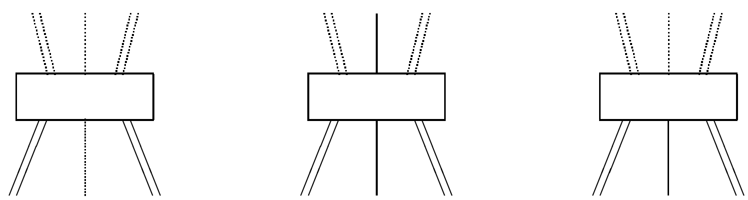
$$= \quad \dfrac{\Gamma_1 \xrightarrow{f} \Gamma_2, A, \Gamma_3 \qquad \dfrac{\Delta_1 \xrightarrow{g} \Delta_2, B, \Delta_3 \qquad \Phi_1, A, \Phi_2, B, \Phi_3 \xrightarrow{h} \Phi_4}{\Phi_1, A, \Phi_2, \Delta_1, \Phi_3 \xrightarrow{g\,;\,h} \Delta_2, \Phi_4, \Delta_3}}{\Phi_1, \Gamma_1, \Phi_2, \Delta_1, \Phi_3 \xrightarrow{f\,;\,(g\,;\,h)} \Delta_2 \mid \Gamma_2, \Phi_4, \Delta_3 \mid \Gamma_3}$$

(5)[inter1]
$$\dfrac{\dfrac{\Gamma_1 \xrightarrow{f} \Gamma_2, A, \Gamma_3, B, \Gamma_4 \qquad \Delta_1, A, \Delta_2 \xrightarrow{g} \Delta_3}{\Delta_1, \Gamma_1, \Delta_2 \xrightarrow{f\,;\,g} \Gamma_2, \Delta_3, \Gamma_3, B, \Gamma_4} \qquad \Phi_1, B, \Phi_2 \xrightarrow{h} \Phi_3}{\Phi_1 \mid \Delta_1, \Gamma_1, \Phi_2 \mid \Delta_2 \xrightarrow{(f\,;\,g)\,;\,h} \Gamma_2, \Delta_3, \Gamma_3, \Phi_3, \Gamma_4}$$

$$= \quad \dfrac{\dfrac{\Gamma_1 \xrightarrow{f} \Gamma_2, A, \Gamma_3, B, \Gamma_4 \quad \Phi_1, B, \Phi_2 \xrightarrow{h} \Phi_3}{\Phi_1, \Gamma_1, \Phi_2 \xrightarrow{f\,;h} \Gamma_2, A, \Gamma_3, \Phi_3, \Gamma_4} \qquad \Delta_1, A, \Delta_2 \xrightarrow{g} \Delta_3}{\Phi_1 \mid \Delta_1, \Gamma_1, \Phi_2 \mid \Delta_2 \xrightarrow{(f\,;h)\,;g} \Gamma_2, \Delta_3, \Gamma_3, \Phi_3, \Gamma_4}$$

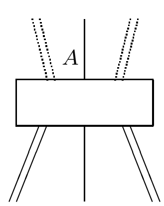
This completes the definition of a polarized polycategory.

4.3. Polarized proof circuits. There is a very convenient visual term logic for polarized polycategories in which the identities become topological identities. This is a generalization of the proof circuits introduced in [BCST96]. The generalization requires that one labels wires not only by their type but also by the category in which their type lives (*i.e.* the opponent or player category). Following the convention that hollow nodes represent opponent positions and solid nodes represent player positions, we shall draw "opponent wires" as dotted and "player wires" as solid. All wires and nodes are supposed to be labelled, although when unnecessary for the purpose at hand, we shall often drop the labels to make the drawing clearer.

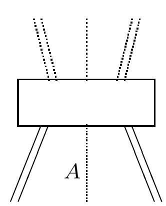


A polyarrow in $\mathbf{X}_o$ A polyarrow in $\mathbf{X}_p$ A polyarrow in $\widehat{\mathbf{X}}$

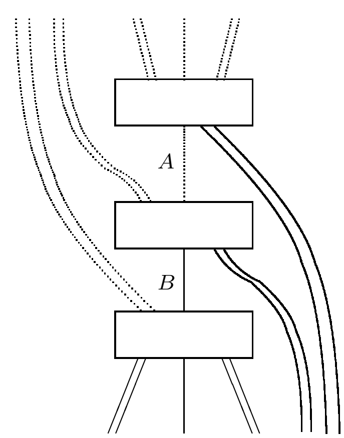
The cut rule is represented by connecting two wires which must be of the same type and polarity. Notice that when a cut is performed the external typing of the circuit will be the same as for the polarized sequent obtained by cutting the two sequents corresponding to the circuits being cut together. The equalities between polyarrows then turn into topological identities. Examples of the circuits involved are easy to construct; all well-typed planar variants are equally valid.



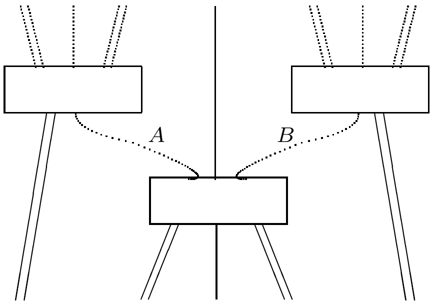
Equation (1)



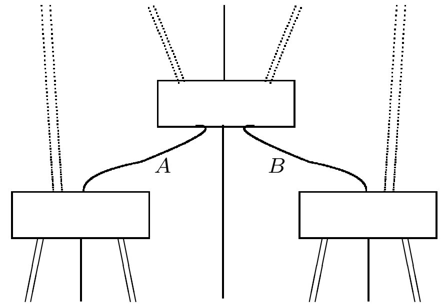
Equation (2)



Equation (3)



Equation (4)



Equation (5)

These circuits allow us to build polarized polycategories based on a given set of polarized types (that is a set of opponent types and a set of player types) and a set of polarized components. The components must each be provided with a type which is a polarized sequent. This data we call a polarized polygraph $\mathcal{G}$ and we can build from it a polarized polycategory $F(\mathcal{G})$ with the same types. We do this by building the polyarrows, which we call polarized proof circuits, inductively from the polarized polygraph by cutting together polarized proof circuits starting with components and wires (the identity maps). Circuit equivalence is then given by topological equivalence of the circuits.

Note that each proof circuit must have an explicit inductive construction from $\mathcal{G}$ using cuts. It is well known that this is equivalent to requiring the result be a planar tree, and indeed in testing that a circuit is a planar tree it is possible to take a rewriting approach (this is often called sequentialization) in which one performs cuts to build larger and larger subcircuits which are proofs. The process is confluent and is successful if the whole circuit is collected [BCST96].

It is clear that polarized polygraphs form a category, **PolyGraph**: the maps are given by maps between the types which preserve polarity and maps between the components which preserve the polarized typing. It is also clear that polarized polycategories form a category **PolPolyCat**: the morphisms are maps between the underlying graphs which, in addition, preserve all the compositions required by the various cut rules (see section 5 for more details).

Every polarized polycategory has an obvious underlying polarized graph and this extends to a functor $U\colon \mathbf{PolPolycat} \longrightarrow \mathbf{PolyGraph}$. We have:

4.3.1. Proposition. *The underling functor* $U\colon \mathbf{PolPolycat} \longrightarrow \mathbf{PolyGraph}$ *has a left adjoint which associates to each polarized polygraph its polarized polycategory of polarized proof circuits.*

The proof of this involves realizing that the notion of topological equivalence is exactly what is given by the associativity and interchange equations of polarized polycategories.

Before looking at some basic examples of polarized polycategories it is worth making some observations about these free polarized polycategories and their relationship to free (unpolarized) polycategories which are constructed in the same manner, see [BCST96].

We shall say a polyarrow is focused in case its type is a player or opponent sequent; it is unfocused otherwise. We start by observing:

4.3.2. Lemma.

1. *A focused polyarrow* $g \in F(\mathcal{G})$ *contains no unfocused components.*

2. An unfocused polyarrow $f \in F(\mathcal{G})$ must contain exactly one unfocused component;

Proof. 1. A focused polyarrow could be a wire or a single component. If it is not one of these it is a polarized cut of two subcircuits. However, note that the focused edge is attached to one of these subcircuits, which means it is focused. Furthermore, the (therefore unfocused) cut edge of this circuit is forced to be a focused edge of the other subcircuit. This means that subcircuits are focused so by induction all subcircuits are focused.

2. If f is unfocused it cannot be a wire. If it consists of a single component we are done. If it is the cut of two subcircuits whichever polarization the cut wire is forces one circuit to be focused (and so by the above has no unfocused components) and the other to be unfocused and thus have, by induction, exactly one unfocused component. ∎

In the terminology of processes, what this is saying is that in a polarized system there can only be one process which is sending a message. This means that the message passing can be regarded as completely sequential. So this does not provide a model of true concurrency.

This observation has some rather surprising formal consequences:

4.3.3. Corollary. *If $\mathcal{G}$ is a polygraph with no unfocused components then $F(\mathcal{G})$ has no mixed polyarrows (*i.e.* the module is empty).*

It is natural to ask whether it is always possible to polarize a polyarrow of a free (unpolarized) polycategory to have the type of a given polarized sequent if we are allowed to have a fresh copy of each component for each possible polarization of that component. Thus, we are asking whether it is possible to rebuild a polyarrow using these new components so that it is a polarization of itself.

4.3.4. Corollary. *In the free polycategory:*

1. *For each polyarrow and focused polarization of its type, there is exactly one polarization of the polyarrow to achieve that polarized type;*

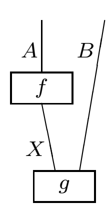
2. *For each polyarrow and unfocused polarization of its type there are exactly n polarizations of the polyarrow, where n is the number of components in the polyarrow.*

Proof. 1. The decomposition of the polyarrow using cuts determines the polarization of subcircuits until the components are reached.

2. Exactly one component must be of mixed polarity; however, we can choose that component freely. On the other hand, once the component is chosen the rest of the typing is determined. ∎

As an illustration, consider the following simple circuit.

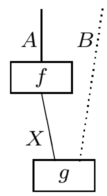
$$\dfrac{A \vdash_f X \quad X, B \vdash_g}{A, B \vdash_{f;g}}$$

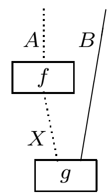


There are only two ways the conclusion may be polarized as a focused sequent: because of the empty conclusion $\perp$ of $f \ ; g$ (which must be typed P), the sequent must have type

P, which can be done two ways, $/\, A \,\backslash\, B \vdash_p \perp$ or $A \,/\, B \,\backslash \vdash_p \perp$. According to the Corollary, for each of these there can be only one way to type the components f, g to achieve the polarization. Indeed, the typing must be done so that exactly one of A or B is of type P, and X must be of the same type as A. This gives these two focused derivations:

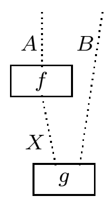
$$\frac{A \vdash_p X \qquad /\, X \,\backslash\, B \vdash_p}{A, B \vdash_p} \qquad \text{and} \qquad \frac{A \vdash_o X \qquad X \,/\, B \,\backslash \vdash_p}{A, B \vdash_p}$$

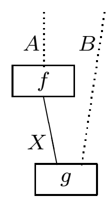
In circuits: 



If we type $f \,;\, g$ as mixed (unfocused), then there are two typings of f, g that accomplish this (note $n = 2$ in this example), one in which X is O, the other in which it is P:

$$\frac{A \vdash_o /\, X \,\backslash \qquad X, B \underset{o\ p}{\mapsto}}{A, B \underset{o\ p}{\mapsto}} \qquad \text{and} \qquad \frac{A \underset{o\ p}{\mapsto} X \qquad /\, X \,\backslash\, B \vdash_p}{A, B \underset{o\ p}{\mapsto}}$$

In circuits: 



4.4. Examples. There are several examples of polarized polycategories, starting with our main example.

4.4.1. Example. Our main example of a polarized polycategory, of course, is derived from the combinatorial AJ games with which we started. To obtain a polarized polycategory from them, we take the same objects as before, but must modify the arrows somewhat to get polyarrows of the appropriate sorts. We shall use the notation from before, adapted to the "multi-channel" context; so each position ("channel") in a polyarrow will be given a "channel name", which will be carried through the formation of new polyarrows. Note that just as before, this notation can also be used to derive a term calculus for polarized polycategories.

[**Opponent polyarrows:**]

$$\alpha \begin{pmatrix} b_1 & \mapsto & h_1 \\ & \cdots & \\ b_m & \mapsto & h_m \end{pmatrix} : \Gamma \longrightarrow \Delta \,/\, \alpha\colon (b_1\colon P_1, \ldots, b_m\colon P_m) \,\backslash\, \Delta'$$

where each $h_i\colon \Gamma \longrightarrow \Delta, \alpha\colon P_i, \Delta'$ is a module polyarrow, and α labels one of the channels.

[**Module polyarrows:**] These are either of the form

$$\overrightarrow{\alpha}[a_k]\cdot g\colon \Gamma \longrightarrow \Delta \,/\, \alpha\colon \{a_1\colon O_1,\ldots,a_n\colon O_n\} \setminus \Delta'$$

where $k \in \{1,\ldots,n\}$, $g\colon \Gamma \longrightarrow \Delta \,/\, \alpha\colon O_k \setminus \Delta'$ is an opponent polyarrow, and α labels one of the channels, or

$$\overleftarrow{\alpha}[b_k]\cdot f\colon \Gamma, \alpha\colon (b_1\colon P_1,\ldots,b_n\colon P_n), \Gamma' \longrightarrow \Delta$$

where $k \in \{1,\ldots,n\}$, $f\colon \Gamma \,/\, \alpha\colon P_k \setminus \Gamma' \longrightarrow \Delta$ is a player polyarrow, and α is a channel label.

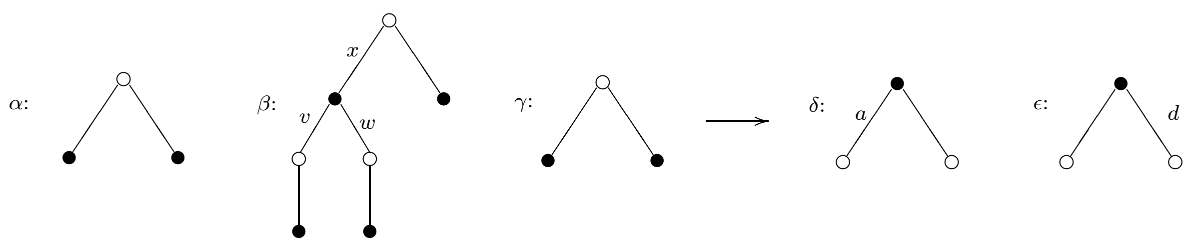
[**Player polyarrows:**]

$$\alpha\left\{\begin{array}{ccc} a_1 & \mapsto & h_1 \\ & \cdots & \\ a_m & \mapsto & h_m \end{array}\right\}\colon \Gamma \,/\, \alpha\colon \{a_1\colon O_1,\ldots,a_m\colon O_m\} \setminus \Gamma' \longrightarrow \Delta$$

where each $h_j\colon \Gamma, \alpha\colon O_j, \Gamma' \longrightarrow \Delta$ is a module polyarrow and α is a channel label.

As an example of such a polyarrow, the following is a module polyarrow. (We've only labeled plays that enter into the construction of the arrow, for clarity.)

$$\overleftarrow{\beta}[x]\cdot\beta\left\{\begin{array}{ccc} v & \mapsto & \overrightarrow{\delta}[a]\cdot\delta() \\ w & \mapsto & \overrightarrow{\epsilon}[d]\cdot\epsilon() \end{array}\right\}:$$



There is a pleasant interpretation of these polyarrows. A polyarrow is a process which we may view as a person, say Mike, sitting in an office with a number of telephones which are directly connected to other offices: some of these telephones are white (and connect to white phones) and some are black (and connect to black phones) . When Mike is a module polyarrow he may pick up any of the telephones and send a message to the person in the office to which that phone is connected. However, Mike can only hold one telephone at a time and, if he sends a message to Mary who is sitting in another office, he must then hold this line until he receives a response from Mary. When he is holding the line in this manner he is "focused" on Mary and is either an opponent or a player polyarrow depending (respectively) on whether the phone he is holding is white or black. Only when Mike has received a response from Mary is he free to put down Mary's phone. At that stage Mike becomes a module polyarrow again and can pick up any of his phones again.

In the example above the conversation went as follows: Mike starts by picking up the white phone, labelled β, and sends the message x to Mary. He then awaits Mary's response which could be, according to the preset protocol, either a v or a w. When Mary responds with a v, Mike will pick up the black phone, labelled δ, and send the message a to Bob. Mike will not expect a response from Bob because his task will be complete and he will never put down the phone. On the other hand, if Mary responds with a w, Mike will pick up the black phone labelled ϵ and send the message d to Martha: he will not expect a response from Martha and so be left holding this phone.

4.4.2. Example. Spans of sets and partial maps form a polarized polycategory $\mathbf{X}$, in the following manner. The objects of $\mathbf{X}_o$,$\mathbf{X}_p$ and $\widehat{\mathbf{X}}$ are all sets; arrows in $\mathbf{X}_o$ are spans $Q_1 \longleftarrow \cdot \rightarrowtail Q_2$ where the right leg is mono, arrows "$P_1 \longrightarrow P_2$" in $\mathbf{X}_p$ are the dual, *i.e.* spans $P_1 \leftarrowtail \cdot \longrightarrow P_2$ where the left leg is mono, and arrows in $\widehat{\mathbf{X}}$ are arbitrary spans. These generalize in a natural way to give polyarrows; we consider cones to all the objects of the sequent with the requirement that (for O and P sequents) the map to the object in focus is a mono. Module polyarrows are arbitrary cones. So, for example, a P sequent would look like this, with m monic.

$$\begin{tikzcd}
 & & & & C \arrow[dllll] \arrow[dll, "m"'] \arrow[d] \arrow[drr] \arrow[drrrr] & & & & \\
Q_1 & \cdots & /\, P \,\backslash & \cdots & Q_n & \vdash_p & P_1 & \cdots & P_n
\end{tikzcd}$$

We may think of this arrow in relational database terms as a table with a specified key. Cut is given by pullback, and it is a standard fact that pullbacks will preserve monics, so the typing is respected.

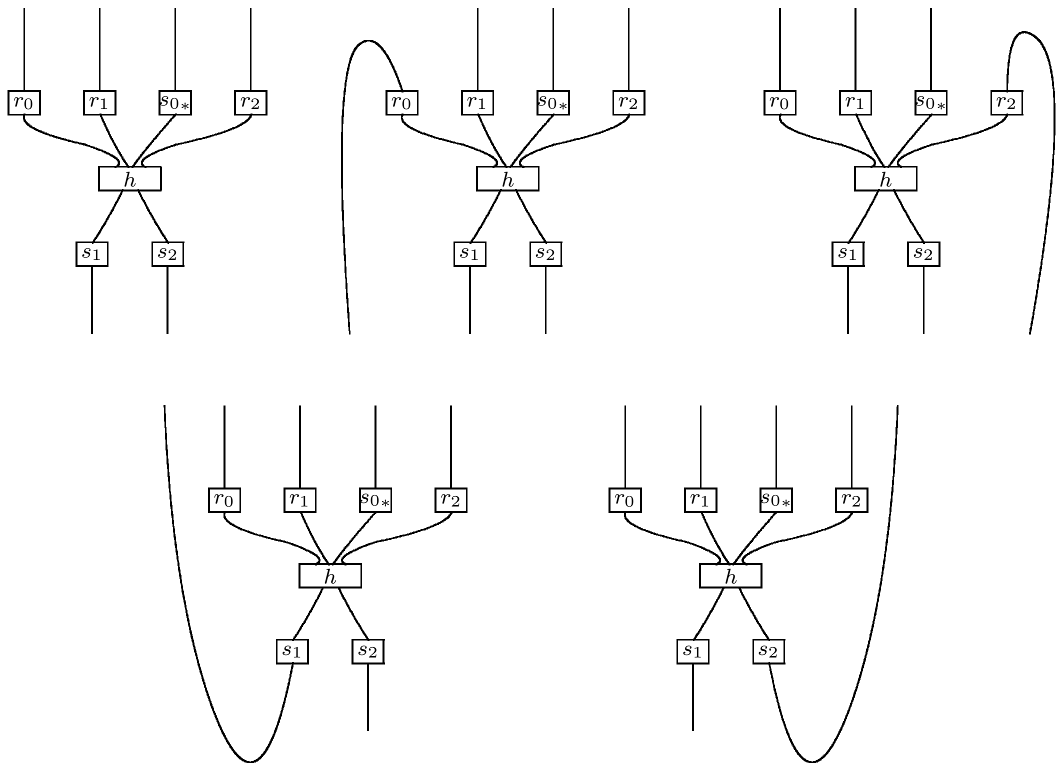
4.4.3. Example. The next example uses the notion of a linear functor between two polycategories; these are introduced in [CS99, CKS]. Let $F = (F_\otimes, F_\oplus)\colon \mathbf{A} \longrightarrow \mathbf{B}$ be a linear functor; then we can obtain a polarized polycategory $\mathbf{X}_F$ from F as follows. The objects of $\mathbf{X}_{F_o}$ are pairs $(A, \otimes)$ for A an object of $\mathbf{A}$; we think of such an object as the object $F_\otimes(A)$. Objects of $\mathbf{X}_{F_p}$ are pairs $(A, \oplus)$, which we shall think of as $F_\oplus(A)$. Note that equality of objects is determined by equality in $\mathbf{A}$, not by equality of images in $\mathbf{B}$. An O-sequent, which we can think of as $F_\otimes(A_1), \cdots, F_\otimes(A_n) \longrightarrow F_\oplus(A'_1), \cdots, F_\oplus(A'_{i-1}) \,/\, F_\otimes(A) \,\backslash\, F_\oplus(A'_{i+1}), \cdots, F_\oplus(A'_m)$, is the $F_\otimes$ image (we called this an $F_\otimes$ functor box in [CS99, CKS]) of an $\mathbf{A}$ sequent $A_1, \cdots, A_n \longrightarrow A'_1, \cdots, A'_{i-1}, A, A'_{i+1}, \cdots, A'_m$. Dually, a P-sequent is the $F_\oplus$ image of an $\mathbf{A}$ sequent. Cross (mixed) sequents are arbitrary $\mathbf{B}$ sequents of the form $F_\otimes(A_1), \cdots, F_\otimes(A_n) \longrightarrow F_\oplus(A'_1), \cdots F_\oplus(A'_m)$. Composition (cut) is given as follows. "Pure" (O or P) sequents compose by composition in $\mathbf{A}$ (taking the appropriate $F_\otimes$ or $F_\oplus$ image, as determined by the typing); composition (cut) with a cross sequent is just composition in $\mathbf{B}$. This is well typed since pure sequents compose with pure sequents to give pure sequents, but composition with a cross sequent gives a cross sequent. The identities for a polycategory are then trivially induced by those identities in the underlying $\mathbf{A}$, $\mathbf{B}$.

We remark here (for the experts) that hypercoherences may be seen as an instance of this example. We recall that a hypercoherence E may be regarded as a "hypergraph", determined by a set $|E|$ of "nodes" and a set $\Gamma(E)$ of "hyperedges", where a hyperedge is a finite non-empty set of nodes; $\Gamma(E)$ is required to contain all singletons (which may be thought of as "loops"). These naturally form a category $\mathbf{HC}$, in fact, a $*$-autonomous category with products (and so coproducts); very roughly, maps may be thought of as relations mapping hyperedges to hyperedges, with the restriction that only loops may be mapped to loops. $\mathbf{HC}$ has a full subcategory $\mathbf{HC}_+$ of "hereditary" hypercoherences, *i.e.* those hypercoherences whose sets of hyperedges are "down-closed", in the sense that if u is a finite non-empty subset of a hyperedge, it also is a hyperedge. $\mathbf{HC}_+$ is a coreflective subcategory of $\mathbf{HC}$; the right adjoint to the inclusion is $\downarrow: \mathbf{HC} \longrightarrow \mathbf{HC}_+$, where $\downarrow E$ has the same nodes as E, but whose hyperedges are the down-closed hyperedges of E, *i.e.* those hyperedges of E all of whose finite non-empty subsets are also hyperedges of E. $\mathbf{HC}_+$ is closed under the tensor and product of $\mathbf{HC}$. Using the duality of $\mathbf{HC}$, we also obtain a full reflective subcategory $\mathbf{HC}_-$ of "anti-hereditary" hypercoherences (those E for which $E^\perp$ is hereditary), which is closed under par and coproduct. The reflection is denoted $\uparrow: \mathbf{HC} \longrightarrow \mathbf{HC}_-$. (See [HS07] for more detail.) The linear functor we have in mind is $(\downarrow, \uparrow)$, whose components are iso-monoidal (with respect to tensor and par respectively). In general, if the linear functor F is a coreflection, reflection pair in this way, the polarized polycategory $\mathbf{X}_F$ is an inner polarized polycategory (its module is given by an adjunction); with hypercoherences, this is indeed the case, so the fact that $\mathbf{HC}$ gives a model of polarity may be interpreted in this way.

4.4.4. Example. Example 3.0.2 can be extended to the polarized polycategory setting, in the following way. Suppose $\mathbf{C}$ is a polycategory, suppose that $\mathcal{K}$ is a class of polyarrows $\mathsf{I}, \cdots, \mathsf{I} \longrightarrow \mathsf{J}, \cdots, \mathsf{J}$ for arbitrarily many I, J in the domain and codomain (respectively), and suppose that I, J are a cyclic linear adjoint pair (effectively, this means one is the "negation" of the other). Suppose that $\mathcal{K}$ is closed under "negating" wires, in the sense that if $\mathsf{I}, \mathsf{I}, \cdots, \mathsf{I}, \mathsf{I} \longrightarrow \mathsf{J}, \mathsf{J}, \cdots, \mathsf{J}, \mathsf{J}$ is in $\mathcal{K}$, then so are $\mathsf{I}, \cdots, \mathsf{I}, \mathsf{I} \longrightarrow \mathsf{J}, \mathsf{J}, \mathsf{J}, \cdots, \mathsf{J}, \mathsf{J}$, and $\mathsf{I}, \mathsf{I}, \cdots, \mathsf{I} \longrightarrow \mathsf{J}, \mathsf{J}, \cdots, \mathsf{J}, \mathsf{J}, \mathsf{J}$, and $\mathsf{I}, \mathsf{I}, \mathsf{I}, \cdots, \mathsf{I}, \mathsf{I} \longrightarrow \mathsf{J}, \cdots, \mathsf{J}, \mathsf{J}$, and $\mathsf{I}, \mathsf{I}, \cdots, \mathsf{I}, \mathsf{I}, \mathsf{I} \longrightarrow \mathsf{J}, \mathsf{J}, \cdots, \mathsf{J}$ all in $\mathcal{K}$, each obtained by "bending" one of the "end wires" (in circuit terms), *i.e.* by bringing one of the end formulas over to the other side, using the linear adjunction between I, J. The orthogonality relation between (ordinary) arrows remains the same. Then $\mathcal{G}(\mathbf{C}, \mathcal{K})$ is a polarized polycategory whose objects are the same as in Example 3.0.2 and whose polyarrows are induced by those in $\mathbf{C}$. For example

$$(\mathcal{R}_0, X_0), (\mathcal{R}_1, X_1) \,/(Y_0, \mathcal{S}_0) \backslash (\mathcal{R}_2, X_2) \xrightarrow{h} (Y_1, \mathcal{S}_1), (Y_2, \mathcal{S}_2)$$

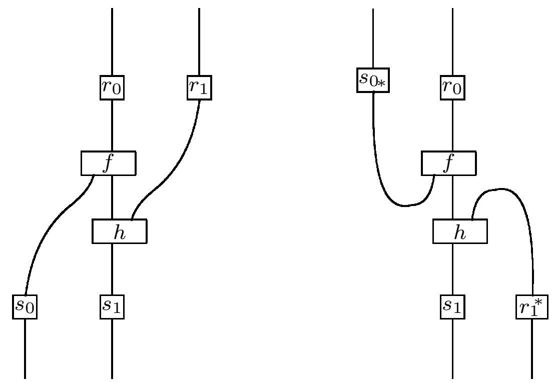
is a P-polyarrow if $X_0, X_1, Y_0, X_2 \xrightarrow{h} Y_1, Y_2$ is a polyarrow of $\mathbf{C}$ so that $r_0 \,;\, r_1 \,;\, s_0' \,;\, r_2 \,;\, h \perp s_1 \,;\, s_2$ for all $r_i \in \mathcal{R}_i$, $s_j \in \mathcal{S}_j$, $s_0' \in \mathcal{S}_{0*}$ $(i = 0, 1, 2,\ j = 1, 2)$. In view of the condition on $\mathcal{K}$, this means all the following are in $\mathcal{K}$, for any r's and s's.



Other polyarrows are defined similarly, using the adjoint operators $\mathcal{S}_*$, $\mathcal{R}^*$ for types in focus.

"Pure" cuts (*i.e.* those not using module polyarrows) are not a problem; the issue is how to cut with module polyarrows, where the condition forces an r to "meet" an r^*, or an s with an s_*. For example, consider a cut between an O-polyarrow $\Gamma_0 \xrightarrow{f} \Delta_0 \,/\, Q \setminus$ and a module polyarrow $Q, \Gamma_1 \xrightarrow{h} \Delta_1$, where $Q = (\mathcal{R}, X)$ is an O-object. The result is a module polyarrow $\Gamma_0, \Gamma_1 \xrightarrow{f;h} \Delta_0, \Delta_1$. Of course, the underlying cut in $\mathbf{C}$ is straightforward; the issue is whether the module polyarrow condition is satisfied. For simplicity, suppose the Γ's and Δ's are singletons. Choose any r_0, r_1, s_0, s_1 of the right types for $\Gamma_0, \Gamma_1, \Delta_0, \Delta_1$ and cut these with $f \,;\, h$: we must show the resulting polyarrow is in $\mathcal{K}$. The idea is simple if expressed in circuit terms: bend the I (respectively J) wires which are inputs for h (respectively outputs for f) out of the way (so they become J output wires for h, respectively I input wires for f), and notice that the subcircuit of this new graph consisting of all wires (including X) into and out of h has the property that for any $r \in \mathcal{R}$, cutting with r produces a polyarrow in $\mathcal{K}$. Hence that subcircuit is a possible r^* for f, and so f cut with that subcircuit is in $\mathcal{K}$.

Straighten out the bent wires, and one concludes that $f\ ;\ g$ is in $\mathcal{K}$.



5. Additive types for polarized polycategories

To get a game logic which has channels we need to combine the basic game logic with the logic of polarized cuts. This means we must describe the additive connectives in the polarized context. However, there are two ways we could do this: using polarized additives and using non-polarized additives. Although there are many similarities, the differences in these two forms of additive structure are striking.

5.1. Polarized additives. The sequent rules for the polarized additives are given in Table 6. Notice that the duality between the left and right column of rules has been maintained. This logic with the twenty four rules for cut satisfies cut elimination and indeed it is not hard to show that cut elimination gives a confluent rewriting system much as in the basic game logic. We shall discuss the polycategorical cut-elimination theorem at the end of the section. We shall start, however, by concentrating on the polarized polycategorical proof theory of the logic.

A polarized polycategory $\mathbf{X}$ has polarized coproducts if for every (finite) family $(X_i)_{i \in I}$, of opponent objects there is an player object $\bigsqcup_{i \in I} X_i$ such that there is a polarized polynatural equivalence in Γ, Γ' and Δ:

$$\prod_{i \in I} \widehat{\mathbf{X}}(\Gamma, X_i, \Gamma'\ ;\ \Delta) \cong \mathbf{X}_p(\Gamma \,/ \textstyle\bigsqcup_{i \in I} X_i \setminus \Gamma'\ ;\ \Delta)$$

To be polynatural in an argument means that any cut through that argument will induce a commuting square so that cutting above the line then moving through the equivalence will be the same as moving through the equivalence and cutting (in parallel) below the line. Notice in particular that corresponding to the identity opponent map on the polarized coproduct, this gives a family of mixed coprojection maps, so the expected structure of a polarized coproduct is implicit in this definition.

Dually, a polarized polycategory has polarized products if for every (finite) family $(Y_j)_{j \in J}$, of player objects there is an opponent object $\prod_{j \in J} Y_j$ such that there is a polarized polynatural

Table 6: General polarized additives

$$\frac{}{/\,A\,\backslash\ \vdash_p A}\quad \textit{atomic identities}\quad \frac{}{A \vdash_o\ /\,A\,\backslash}$$

$$\frac{\left\{\Gamma, X_i, \Gamma' \underset{o\ p}{\vdash\!\!\!\rightarrow} \Delta\right\}_{i\in I}}{\Gamma\,/\bigsqcup_{i\in I} X_i\,\backslash\,\Gamma' \vdash_p \Delta}\ \textit{cotuple}\qquad \frac{\left\{\Gamma \underset{o\ p}{\vdash\!\!\!\rightarrow} \Delta, Y_i, \Delta'\right\}_{i\in I}}{\Gamma \vdash_o \Delta\,/\prod_{i\in I} Y_i\,\backslash\,\Delta'}\ \textit{tuple}$$

$$\frac{\Gamma \vdash_o \Delta\,/\,X_k\,\backslash\,\Delta'}{\Gamma \underset{o\ p}{\vdash\!\!\!\rightarrow} \Delta, \bigsqcup_{i\in I} X_i, \Delta'}\ \textit{injection}\qquad \frac{\Gamma\,/\,Y_k\,\backslash\,\Gamma' \vdash_p \Delta}{\Gamma, \prod_{i\in I} Y_i, \Gamma' \underset{o\ p}{\vdash\!\!\!\rightarrow} \Delta}\ \textit{projection}$$

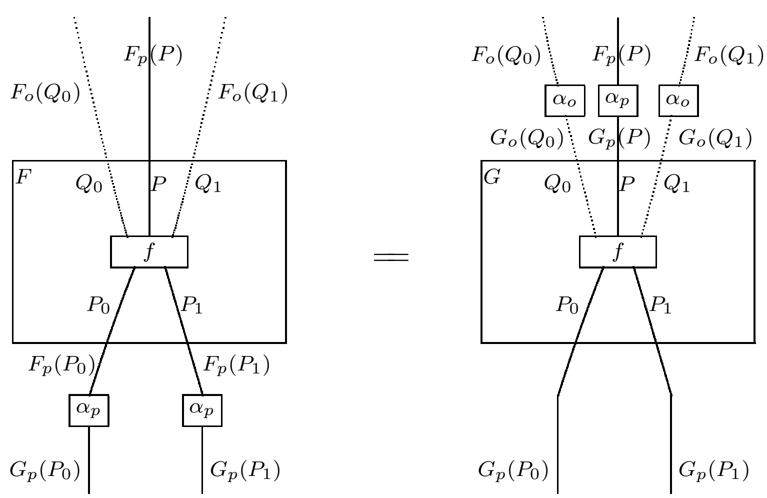
where $k \in I$, $I \neq \emptyset$

equivalence in Γ, Δ and Δ':

$$\prod_{j\in J} \widehat{\mathbf{X}}(\Gamma\ ;\ \Delta, Y_i, \Delta') \cong \mathbf{X}_o(\Gamma\ ;\ \Delta\,/\textstyle\prod_{j\in J} Y_i\,\backslash\,\Delta')$$

A polarized polycategory with polarized products and coproducts is called a polarized game polycategory. It is immediately obvious that combinatorial AJ games (Example 4.4.1) are an example of a polarized game polycategory.

We have already mentioned that there is a natural notion of a polarized polyfunctor between polarized polycategories: this corresponds to a polarized version of what was called a morphism (by analogy with the bicategorical notion) in [CKS]. There is in addition a natural notion of polarized polytransformation, $\alpha\colon F \longrightarrow G$, consisting of two natural transformations $\alpha_o\colon F_o \longrightarrow G_o$ and $\alpha_p\colon F_p \longrightarrow G_p$, satisfying the condition that for any polyarrow f, postcomposing $F(f)$ with appropriate α's equals precomposing $G(f)$ with appropriate α's. There are several variants of this, depending on the type of f. We illustrate the general idea with a P polyarrow.



Thus, polarized polycategories form a 2-category, which we denote by **PolPolyCat**. Similarly polarized game polycategories form a 2-category **PolPolyGam** with polarized polyfunctors which preserve all polarized products and coproducts and polarized poly transformations. There is an obvious underlying functor from these polarized game polycategories to polarized polycategories. As before we observe:

5.1.1. PROPOSITION. *There is a 2-adjunction*

$$\mathbf{PolPolyCat} \underset{\mathsf{Gam}}{\overset{\mathsf{U}}{\underset{\longrightarrow}{\overset{\longleftarrow}{\top}}}} \mathbf{PolPolyGam}$$

PROOF. (Sketch) The main difficulty is the construction of $\mathsf{Gam}(\mathbf{X})$ for an arbitrary polarized polycategory $\mathbf{X}$. The objects of $\mathsf{Gam}(\mathbf{X})$ are the formal games on objects in $\mathbf{X}$. The maps are defined inductively on the size of the sequent which is the type of the map.

For a player sequent the object in focus must either be a polarized coproduct or an object from $\mathbf{X}$. In the former case the maps must be a cotuple of mixed polyarrows (with smaller types) and so it is determined inductively. In the latter case there are no maps unless the other objects are also all from $\mathbf{X}$, in which case the maps are precisely those in $\mathbf{X}$. The dual arrangement hold for opponent maps.

This leaves polyarrows with mixed sequents as types. In this case either the sequent is from $\mathbf{X}$ or there must be at least one polarized product type on the left or polarized coproduct type on the right. In the former case the polyarrows are exactly those in $\mathbf{X}$. In the latter case we must consider all the focused "derivatives" of this mixed sequent. These are the focused sequents obtained from the mixed sequent by choosing a component of one of the composite polarized product or coproduct types and projecting or coprojecting to it. The maps of the original sequent are the disjoint union of the maps of the derivatives.

Finally composition is defined by following the cut elimination steps (in particular projection of tuple and coprojection of cotuple) until polyarrows from $\mathbf{X}$ are encountered at which stage one uses the cut defined in $\mathbf{X}$. ∎

We can extract a Whitman theorem from this: given a polarized game polycategory we shall say that an object is player atomic in case whenever it is the focus of a player sequent in which some other object is a polarized product or coproduct then the poly-hom-set is empty. Similarly one can define opponent atomic objects as those which when they are in focus always have the poly-hom-set empty if there is a polarized product or coproduct involved.

Notice that atomic objects cannot be themselves polarized products or coproducts or else they would not have identity maps.

We then say a polycategory is soft in case every mixed sequent map which involves at least one polarized coproduct or product factors through a unique projection or coprojection. A polarized game category is said to be Whitman in case every object is a game of atoms and the polycategory is soft. We then have:

5.1.2. PROPOSITION. *A polarized game polycategory is equivalent to* $\mathsf{Gam}(\mathbf{X})$ *for some polarized polycategory* $\mathbf{X}$ *if and only if it is Whitman.*

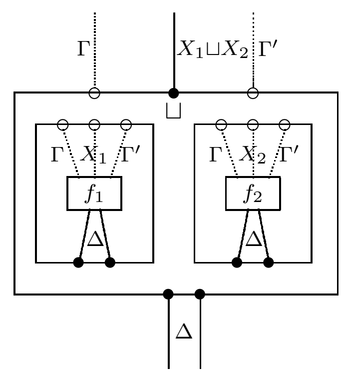
As before when $\mathbf{Y}$ is a Whitman polarized game category the inclusion functor of the atoms into $\mathbf{Y}$ has the 2-universal property for the underling functor from polarized game polycategories to polarized categories.

Implicit in the Whitman theorem is the polycategorical cut-elimination procedure for this logic. We shall now illustrate the key steps using the circuit representation for terms introduced earlier.

5.2. Proof circuits for polarized additives. To extend the proof circuits to include the polarized additives, one can use a simple modifications of the idea of "proof boxes", originally due to Girard and adapted by us for linear functors [CS99]. Given several subcircuits which have the same labels on their input and output wires, except for one input (in the case of $\sqcup$), or one output (in the case of $\sqcap$), which varies from one box to the next, we can "box" them together into a larger circuit which has those same labelled input and output wires, with the variable port now labelled by the $\sqcup$ of the varying labels (or, in the dual case, the $\sqcap$ of them). Each subcircuit is put into a box, with its input and output wires attached to a series of nodes (called "ports"). Note that each such box will have the same labelled ports, apart from the one that varies. We use the usual convention of open nodes for "opponent" nodes and solid ones for "player" nodes. All these boxes are then surrounded with a large box bearing the same labelled ports, apart from the one now bearing the $\sqcup$ (or $\sqcap$) label. We shall usually indicate that port with a $\sqcup$ (or $\sqcap$), although that may be dropped since the port in question is the only solid (or open for $\sqcap$) port among the inputs (outputs for $\sqcap$). (We may at times drop the other labels, with the understanding that the left-right order of the ports represents the common labels.)

Here is an example of the box operation, corresponding to the displayed derivation.

$$\frac{f_1 :: \Gamma, X_1, \Gamma' \underset{o\;\;p}{\mapsto} \Delta \qquad f_2 :: \Gamma, X_2, \Gamma' \underset{o\;\;p}{\mapsto} \Delta}{\alpha\{f_1, f_2\} :: \Gamma \,/\, X_1 \sqcup X_2 \,\backslash\, \Gamma' \vdash_p \Delta}$$



In addition we shall need "injection" and "projection" components (think of these as the obvious axioms); the injection and projection rules may be derived from these *via* cut. An injection component is illustrated in the second example below. We remark that giving projections and injections in terms of components, rather than using boxes as we do for the tupling and cotupling rules, actually has some computational content. Really what the sequent calculus rules suggest literally would involve "scope boxes" for the projections and injections, since they are given by terms, not types. So the b_k and p_k components are effectively "principal ports" of such "scope boxes". However, among the term rewrite rules (equivalences on terms) are rules which admit of enlarging the "scope" as much as typing permits, in effect allowing one to always assume the maximal scope. So in effect, scope becomes unnecessary, and we can gain some simplicity in the rules by ignoring it in the circuits.

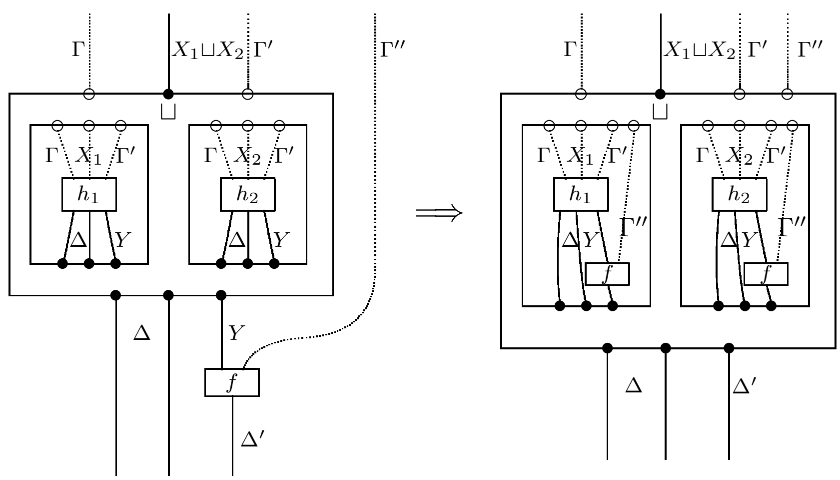
To deal with the units (initial and terminal objects) we need only use the nullary cases of these boxes. So for example, an empty $\sqcup$ box with arbitrary (dotted) inputs Γ, Γ' and arbitrary (solid) outputs Δ, as well as a (solid) $\mathbf{0}$ input between the Γ, Γ', is the circuit for the axiom (derivation with no hypothesis) $\Gamma, \mathbf{0}, \Gamma' \vdash_p \Delta$. The dual gives the $\mathbf{1}$ axiom or derivation. (These are in fact unique, under the equivalence given by the rewrites we describe next.)

There are a number of rewrites associated with these boxes which are the analogue of the rewrites in section 2.1. They are precisely the cut elimination steps for the logic. We provide two reduction steps below and once their duals are added this provides the complete set for this graphical representation. We also include the term representation of these reductions. The system also requires expansion rules for the additive types which are illustrated as well.

$\alpha\{a_i \mapsto h_i\}_i \; ; \; f \Rightarrow \alpha\{a_i \mapsto h_i \; ; \; f\}_i$

$$\frac{\dfrac{h_1 :: \Gamma, X_1, \Gamma' \underset{o\;p}{\mapsto} \Delta, Y \qquad h_2 :: \Gamma, X_2, \Gamma' \underset{o\;p}{\mapsto} \Delta, Y}{\alpha\{h_1, h_2\} :: \Gamma \,/\, X_1 \sqcup X_2 \,\backslash\, \Gamma' \vdash_p \Delta, Y} \qquad f :: \;/\, Y \,\backslash\, \Gamma'' \vdash_p \Delta'}{\alpha\{h_1, h_2\} \; ; \; f :: \Gamma \,/\, X_1 \sqcup X_2 \,\backslash\, \Gamma', \Gamma'' \vdash_p \Delta, \Delta'} \qquad \Longrightarrow$$

$$\frac{\dfrac{h_1 :: \Gamma, X_1, \Gamma' \underset{o\;p}{\mapsto} \Delta, Y \qquad f :: \;/\, Y \,\backslash\, \Gamma'' \vdash_p \Delta'}{h_1 \; ; \; f :: \Gamma, X_1, \Gamma', \Gamma'' \underset{o\;p}{\mapsto} \Delta, \Delta'} \qquad \dfrac{h_2 :: \Gamma, X_2, \Gamma' \underset{o\;p}{\mapsto} \Delta, Y \qquad f :: \;/\, Y \,\backslash\, \Gamma'' \vdash_p \Delta'}{h_2 \; ; \; f :: \Gamma, X_2, \Gamma', \Gamma'' \underset{o\;p}{\mapsto} \Delta, \Delta'}}{\alpha\{h_1 \; ; \; f, h_2 \; ; \; f\} :: \Gamma \,/\, X_1 \sqcup X_2 \,\backslash\, \Gamma', \Gamma'' \vdash_p \Delta, \Delta'}$$

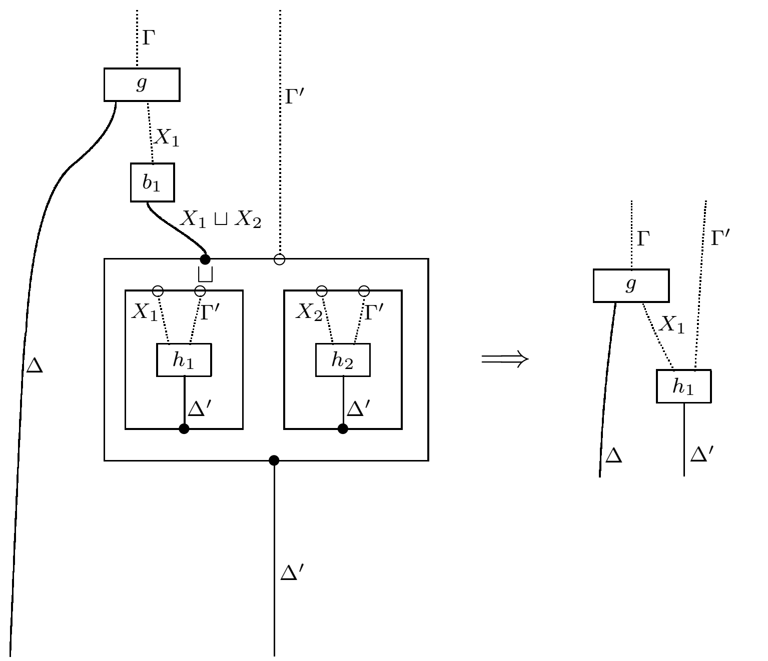


There is a dual rewrite allowing the $\sqcup$ box to "absorb" a component "above" it as well; of course there are dual rules for $\sqcap$ boxes.

$\overrightarrow{\alpha}[a_k] \cdot g \; ; \alpha\{a_i \mapsto h_i\}_i \Rightarrow g \; ; \; h_k$

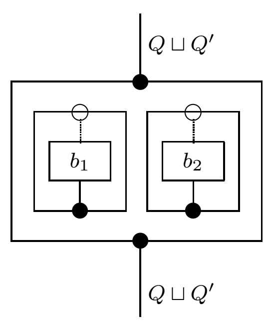
$$\frac{\dfrac{g :: \Gamma \vdash_o \Delta \,/\, X_1 \,\backslash}{\overrightarrow{\alpha}[a_1] \cdot g :: \Gamma \underset{o\;p}{\mapsto} \Delta, X_1 \sqcup X_2} \qquad \dfrac{h_1 :: X_1, \Gamma' \underset{o\;p}{\mapsto} \Delta' \qquad h_2 :: X_2, \Gamma' \underset{o\;p}{\mapsto} \Delta'}{\alpha\{h_1, h_2\} :: \;/\, X_1 \sqcup X_2 \,\backslash\, \Gamma' \vdash_p \Delta'}}{\overrightarrow{\alpha}[a_1] \cdot g \; ; \; \alpha\{h_1, h_2\} :: \Gamma, \Gamma' \underset{o\;p}{\mapsto} \Delta, \Delta'}$$

$$\Longrightarrow \qquad \frac{g :: \Gamma \vdash_o \Delta \,/\, X_1 \,\backslash \qquad h_1 :: X_1, \Gamma' \underset{o\;\,p}{\mapsto} \Delta'}{g \,;\, h_1 :: \Gamma, \Gamma' \underset{o\;\,p}{\mapsto} \Delta, \Delta'}$$



There are dual rewrites for $\sqcap$ boxes.

There are expansion rewrites as well; for example the identity on $Q \sqcup Q'$ expands to this circuit.



5.2.1. Proposition. *The four families of reductions and two expansions form a confluent expansion/reduction system on the circuits for polarized game polycategories.*

We shall not prove this here but rather refer the interested reader to Craig Pastro's thesis [P03] in which a similar (but slightly more complex) system is discussed.

5.3. Non-polarized additives. A polarized polycategory can also have additives which are not polarized, more precisely, which do not change polarity, in two ways in fact. We shall consider additives which have the coproducts in the player category and the products in the opponent category, and the reverse, with coproducts in the opponent category and products in the player category. The point of these two notions, of course, is that these additives do

Table 7: General aft additives

$$\frac{\{\Gamma \,/\, X_i \setminus \Gamma' \vdash_p \Delta\}_{i\in I}}{\Gamma \,/\, \bigvee_{i\in I} X_i \setminus \Gamma' \vdash_p \Delta}\ \mathit{cotuple} \qquad \frac{\{\Gamma \vdash_o \Delta \,/\, Y_i \setminus \Delta'\}_{i\in I}}{\Gamma \vdash_o \Delta \,/\, \bigwedge_{i\in I} Y_i \setminus \Delta'}\ \mathit{tuple}$$

$$\frac{\Gamma \vdash_p \Delta, X_k, \Delta'}{\Gamma \vdash_p \Delta, \bigvee_{i\in I} X_i, \Delta'}\ \mathit{injection0} \qquad \frac{\Gamma, Y_k, \Gamma' \vdash_o \Delta}{\Gamma, \bigwedge_{i\in I} Y_i, \Gamma' \vdash_o \Delta}\ \mathit{projection0}$$

$$\frac{\Gamma \vdash_o \Delta, X_k, \Delta' \,/\, Y \setminus \Delta''}{\Gamma \vdash_o \Delta, \bigvee_{i\in I} X_i, \Delta' \,/\, Y \setminus \Delta''}\ \mathit{injection1a} \qquad \frac{\Gamma, Y_k, \Gamma' \,/\, X \setminus \Gamma'' \vdash_p \Delta}{\Gamma, \bigwedge_{i\in I} Y_i, \Gamma' \,/\, X \setminus \Gamma'' \vdash_p \Delta}\ \mathit{projection1a}$$

$$\frac{\Gamma \vdash_o \Delta \,/\, Y \setminus \Delta', X_k, \Delta''}{\Gamma \vdash_o \Delta \,/\, Y \setminus \Delta', \bigvee_{i\in I} X_i, \Delta''}\ \mathit{injection1b} \qquad \frac{\Gamma \,/\, X \setminus \Gamma', Y_k, \Gamma'' \vdash_p \Delta}{\Gamma \,/\, X \setminus \Gamma', \bigwedge_{i\in I} Y_i, \Gamma'' \vdash_p \Delta}\ \mathit{projection1b}$$

$$\frac{\Gamma \underset{o\ p}{\mapsto} \Delta, X_k, \Delta'}{\Gamma \underset{o\ p}{\mapsto} \Delta, \bigvee_{i\in I} X_i, \Delta'}\ \mathit{injection2} \qquad \frac{\Gamma, Y_k, \Gamma' \underset{o\ p}{\mapsto} \Delta}{\Gamma, \bigwedge_{i\in I} Y_i, \Gamma' \underset{o\ p}{\mapsto} \Delta}\ \mathit{projection2}$$

where $k \in I$, $I \neq \emptyset$

not alter the polarity, unlike the polarized additives. (The astute reader who notices that one possibility is missing, *viz.* an alternate polarized notion which, for example, forms a "coproduct" of player objects as an opponent object, will also note that this notion doesn't type well, for example, making impossible injections and projections.)

We shall call the first non-polarized additives "aft" additives, the second being "fore" additives. The fore additives are those considered by Olivier Laurent; we shall therefore elaborate the structure of the aft additives first, as they may be less familiar.

One of our main motivations for studying the non-polarized additives is that they have an important role in the representation theorems we shall develop. However, as we shall shortly see, they also have an interesting process interpretation when combined with requiring that the polarized category be "inner" in the spirit of having an inner adjoint to the identity functor.

5.3.1. Aft additives. In Table 7 the sequent rules for the aft additives are provided; notice that these rules maintain the opponent player duality. Following the pattern set by the polarized additives we say that a polarized polycategory has aft sums in case for any finite family $(X_i)_{i\in I}$ of player objects, there is an player object $\bigvee_{i\in I} X_i$ such that there is a polarized polynatural equivalence in Γ, Γ' and Δ:

$$\prod_{i\in I} \mathbf{X}_p(\Gamma \,/\, X_i \setminus \Gamma' \,;\, \Delta) \cong \mathbf{X}_p(\Gamma \,/\, \textstyle\bigvee_{i\in I} X_i \setminus \Gamma' \,;\, \Delta)$$

where the aft coproduct is in focus. The definition of aft products $\bigwedge_{j\in J} Y_j$ is dual.

5.3.2. Fore additives. The characterization of fore sums is similar (the sequent rules are summarized in Table 8): a polarized polycategory has fore sums in case for any finite family $(X_i)_{i\in I}$ of opponent objects, there is an opponent object $\sum_{i\in I} X_i$ such that there is a polarized polynatural equivalence in Γ, Γ', Δ and Δ':

$$\prod_{i\in I} \mathbf{X}_o(\Gamma, X_i, \Gamma' \,;\, \Delta \,/\, Y \setminus \Delta') \cong \mathbf{X}_o(\Gamma, \textstyle\sum_{i\in I} X_i, \Gamma' \,;\, \Delta \,/\, Y \setminus \Delta')$$

with similar such equivalences in $\widehat{\mathbf{X}}$ and $\mathbf{X}_p$, (as suggested by the other cotuple rules in Table 8). The definition for fore products $\prod_{j\in J} Y_j$ is dual.

5.3.3. Remark. Note that the fore sum is not in focus in the cotuple rules (nor is the fore product in focus in the tuple rules), in contrast to the aft additives. However, the fore additives are in focus in the injection and projection rules, again in contrast to the aft additives. This is how one can immediately see that the fore additives provide the semantics for Laurent's additives.

For binary fore sums and fore products, we shall use the notation $Q+Q'$ and $P\times P'$. We might remark that although we seem to have a lot of sequent rules for these additives, the reader should keep in mind that essentially these are the familiar sequent rules for sums and products, with the additional matter of typing the sequents correctly to obtain the various polarizations.

To help recall our terminology, we note the following equations (recall Lemma 3.2.3, for the equations involving the aft additives; the equations for the fore additives are easily proved using the universal properties, as sketched below). To create polarized additives, one may use an aft additive *aft*er a polarity switch, whereas one would use a fore additive be*fore* a polarity switch:

$$\textstyle\bigvee_I Q_i{}^* = \bigsqcup_I Q_i = (\sum_I Q_i)^* \quad \text{and} \quad \bigwedge_J P_{j*} = \bigsqcap_J P_j = (\prod_J P_j)_*$$

Proof: For the fore additives we shall sketch a simple proof that $\bigsqcup_I Q_i = (\sum_I Q_i)^*$.

$$\dfrac{\dfrac{\dfrac{\Gamma \,/ (\sum_I Q_i)^* \setminus \Gamma' \vdash_p \Delta}{\Gamma, \sum_I Q_i, \Gamma' \underset{o\;p}{\vdash\!\!\rightarrow} \Delta}}{\{\Gamma, Q_i, \Gamma' \underset{o\;p}{\vdash\!\!\rightarrow} \Delta\}_i}}{\Gamma \,/ \bigsqcup_I Q_i \setminus \Gamma' \vdash_p \Delta}$$

5.4. Adding aft additives freely. The proof theory of aft products and coproducts is not as well-behaved as polarized products and coproducts: the fact that they are non-polarized permits projections and coprojections to occur outside the focus in focused sequents and introduces commuting conversions into the proof theory and equalities into the rewriting system. We shall discuss a term logic for a more complex system at the end of the section in which these commuting conversions are presented.

There is an alternative way to describe the free category with aft products and coproducts. Notice that categories with aft products and coproducts form a 2-category with polyfunctors which preserve aft products and coproducts as 1-cells, and poly transformations as 2-cells, which we call $\mathbf{PolPolyCat}_{\vee\wedge}$. There is then an evident underlying 2-functor from this 2-category to the 2-category of polarized polycategories with a left 2-adjoint.

Table 8: General fore additives

$$\frac{\{\Gamma, X_i, \Gamma' \vdash_o \Delta \,/\, Y \,\backslash\, \Delta'\}_{i\in I}}{\Gamma, \sum_{i\in I} X_i, \Gamma' \vdash_o \Delta \,/\, Y \,\backslash\, \Delta'}\ \mathit{cotuple0} \qquad \frac{\{\Gamma \,/\, X \,\backslash\, \Gamma' \vdash_p \Delta, Y_i, \Delta'\}_{i\in I}}{\Gamma \,/\, X \,\backslash\, \Gamma' \vdash_p \Delta, \prod_{i\in I} Y_i, \Delta'}\ \mathit{tuple0}$$

$$\frac{\{\Gamma, X_i, \Gamma' \,/\, Y \,\backslash\, \Gamma'' \vdash_p \Delta\}_{i\in I}}{\Gamma, \sum_{i\in I} X_i, \Gamma' \,/\, Y \,\backslash\, \Gamma'' \vdash_p \Delta}\ \mathit{cotuple1} \qquad \frac{\{\Gamma \vdash_o \Delta \,/\, X \,\backslash\, \Delta', Y_i, \Delta''\}_{i\in I}}{\Gamma \vdash_o \Delta \,/\, X \,\backslash\, \Delta', \prod_{i\in I} Y_i, \Delta''}\ \mathit{tuple1}$$

$$\frac{\{\Gamma \,/\, Y \,\backslash\, \Gamma', X_i, \Gamma'' \vdash_p \Delta\}_{i\in I}}{\Gamma \,/\, Y \,\backslash\, \Gamma', \sum_{i\in I} X_i, \Gamma'' \vdash_p \Delta}\ \mathit{cotuple2} \qquad \frac{\{\Gamma \vdash_o \Delta, Y_i, \Delta' \,/\, X \,\backslash\, \Delta''\}_{i\in I}}{\Gamma \vdash_o \Delta, \prod_{i\in I} Y_i, \Delta' \,/\, X \,\backslash\, \Delta''}\ \mathit{tuple2}$$

$$\frac{\left\{\Gamma, X_i, \Gamma' \underset{o\ p}{\mapsto} \Delta\right\}_{i\in I}}{\Gamma, \sum_{i\in I} X_i, \Gamma' \underset{o\ p}{\mapsto} \Delta}\ \mathit{cotuple3} \qquad \frac{\left\{\Gamma \underset{o\ p}{\mapsto} \Delta, Y_i, \Delta'\right\}_{i\in I}}{\Gamma \underset{o\ p}{\mapsto} \Delta, \prod_{i\in I} Y_i, \Delta'}\ \mathit{tuple3}$$

$$\frac{\Gamma \vdash_o \Delta \,/\, X_k \,\backslash\, \Delta'}{\Gamma \vdash_o \Delta \,/\, \sum_{i\in I} X_i \,\backslash\, \Delta'}\ \mathit{injection} \qquad \frac{\Gamma \,/\, Y_k \,\backslash\, \Gamma' \vdash_p \Delta}{\Gamma \,/\, \prod_{i\in I} Y_i \,\backslash\, \Gamma' \vdash_p \Delta}\ \mathit{projection}$$

where $k \in I$, $I \neq \emptyset$

5.4.1. Proposition. *There is a 2-adjunction*

$$\begin{tikzcd} \mathbf{PolPolyCat} \arrow[r, bend right, "\mathsf{F}_{\vee\wedge}"'] \arrow[r, phantom, "\top"] & \mathbf{PolPolyCat}_{\vee\wedge} \arrow[l, bend right, "\mathsf{U}"'] \end{tikzcd}$$

Proof. We start by adding formal finite aft products and coproducts. This can be done (in one shot) by adding coproducts of all finite families of player objects and products of all finite families of opponent objects. One may think of these as maps from finite index sets to the objects of respectively the player and opponent categories. Next we define the polymaps: a player map can be written in the following form (in which certain index sets could be singletons):

$$\bigwedge_{i\in I_1} X_{i,1} \ldots / \bigvee_{j\in J} Y_j \,\backslash \ldots, \bigwedge_{i\in I_n} X_{i,n} \xrightarrow{f} \bigvee_{j\in J_1} Y_{j,1}, \ldots, \bigvee_{j\in J_m} Y_{j,m}$$

and is given by a maps $\check{f}\colon J \longrightarrow \prod_{u=1}^{n} I_u$ and $\breve{f}\colon J \longrightarrow \prod_{v=1}^{m} J_v$ together with a map to polyarrows $f\colon J \longrightarrow \mathbf{X}_p$ where the latter picks out a polyarrow

$$X_{\check{f}(j)_1,1}, \ldots / Y_j \,\backslash \ldots, X_{\check{f}(j)_n,n} \xrightarrow{f(j)} Y_{\breve{f}(j)_1,1}, \ldots, Y_{\breve{f}(j)_m,m}$$

A cut between two such polyarrows must happen at the focus of the second polyarrow, say at "channel index" w. The index functions of the composite are given by composition:

$$\begin{tikzcd} & & J \arrow[dll, "\check{f}_1"'] \arrow[d, "\check{f}_w"] \arrow[drr, "\check{f}_m"] & & \\ I_1 & \cdots & J_w \arrow[dl, "\check{g}_1"'] \arrow[dr, "\check{g}_m"] & \cdots & J_m \\ & I'_1 & \cdots & J'_m & \end{tikzcd}$$

while the associated polyarrow is given by the polarized cut at that coordinate. The opponent category is defined in a dual manner.

The mixed polyarrows

$$\bigwedge_{i\in I_1} X_{i,1} \ldots, \bigwedge_{i\in I_n} X_{i,n} \xrightarrow{h} \bigvee_{j\in J_1} Y_{j,1}, \ldots, \bigvee_{j\in J_m} Y_{j,m}$$

is given by an element $(i_1, \ldots, i_n \,;\, j_1, \ldots, j_m)$ of the product of the index sets together with a mixed polyarrow

$$X_{i_1} \ldots, X_{i_n} \xrightarrow{h} Y_{j_1}, \ldots, Y_{j_m}$$

The cut of a player polyarrow with a mixed polyarrow at channel r is provided by using the element i_r to obtain elements in the index sets of the player elements at the same time as performing the same shaped cut with the associated polyarrows in $\mathbf{X}$.

It is straightforward to check this polarized polycategory has aft products and coproducts and has the 2-universal property with respect to the embedding of $\mathbf{X}$ into it. ∎

Notice that the initial polarized polycategory $F_{\vee\wedge}(\mathbf{0})$ has a discrete polarized category with one player object and one opponent object. Of a little more interest is the free polarized polycategory with products and coproducts generated by a discrete polarized polycategory $\mathbf{T}$. The discrete objects of $\mathbf{T}_p$ may be regarded as free player types and the discrete object $\mathbf{T}_o$ as free opponent types. A player object of $F_{\vee\wedge}(\mathbf{T})$ is a finitely indexed map $X\colon I_X \longrightarrow \mathbf{T}_p$ and similarly an opponent object is a finitely index map $Y\colon J_Y \longrightarrow \mathbf{T}_o$. As there are no generating polyarrows, however, there are only ordinary arrows and no arrows at all in the module. This means that $F_{\vee\wedge}(\mathbf{T})_o$ is the category with free products on opponent types and dually $F_{\vee\wedge}(\mathbf{T})_p$ is the category with free coproducts on player types.

In fact, in general, if the initial polarized polycategory does not have non-trivial polyarrows (which are mixed) the resulting free polarized polycategory will not have non-trivial polyarrows (which are mixed). To obtain a non-trivial polarized polycategory we need to start with one:

5.4.2. Example. *The free polarized polycategory* $F_{\vee\wedge}(\mathbf{1})$

Consider the final polarized polycategory $\mathbf{1}$ and the free polarized category with aft additives constructed from it, $F_{\vee\wedge}(\mathbf{1})$. We may describe it as follows: both player and opponent objects are finite sets. A player map

$$X_1, \ldots / \, Y \setminus \ldots, X_n \xrightarrow{f} Y_1, \ldots, Y_m$$

Table 9: Focusing operations

$$\frac{\Gamma, X, \Gamma' \underset{o\ p}{\mapsto} \Delta}{\Gamma \,/\, X^* \setminus \Gamma' \vdash_p \Delta} \qquad \frac{\Gamma \underset{o\ p}{\mapsto} \Delta, Y, \Delta'}{\Gamma \vdash_o \Delta \,/\, Y_* \setminus \Delta'}$$

$$\frac{\Gamma \vdash_o \Delta \,/\, X \setminus \Delta'}{\Gamma \underset{o\ p}{\mapsto} \Delta, X^*, \Delta'} \qquad \frac{\Gamma \,/\, Y \setminus \Gamma' \vdash_p \Delta}{\Gamma, Y_*, \Gamma' \underset{o\ p}{\mapsto} \Delta}$$

is given by a fan with apex Y with legs $f_i\colon Y \longrightarrow X_i$ and $g_j\colon Y \longrightarrow Y_j$, which we may write as a sequent

$$f_1\colon X_1, \ldots \,/\, Y \setminus \ldots, f_n'\colon X_n \vdash g_1\colon Y_1, \ldots, g_m\colon Y_m$$

Cutting these fans is given by stacking:

$$\frac{f_1\colon X_1 \,/\, Y \setminus f_2\colon X_2 \longrightarrow f_0'\colon Y_0, f_1'\colon Y_1 \quad g_1\colon X_1' \,/\, Y_0 \setminus \longrightarrow g_1'\colon Y_1', g_2'\colon Y_2'}{f_0' \,;\, g_1\colon X_1', f_1\colon X_1 \,/\, Y \setminus f_2\colon X_2 \longrightarrow f_0' \,;\, g_1'\colon Y_1', f_0' \,;\, g_2'\colon Y_2', f_1'\colon Y_1}$$

An opponent map is exactly the opposite of a player map (*i.e.* the opponent polycategory is the opposite of the player polycategory).

On the other hand a mixed polyarrow is given by an element

$$x_1\colon X_1, \ldots, x_n\colon X_n \longrightarrow y_1\colon Y_1, \ldots, y_m\colon Y_m$$

Composition of a mixed polyarrow with a focused polyarrow is given by evaluating the focused arrow at the element associated with the cut formula to obtain elements at all the sets of the resulting mixed polyarrow.

As we shall shortly see this polarized polycategory is closed and is $\otimes$ and $\oplus$-representable.

5.5. Inner polarized polycategories and additives. We are now going to consider a particular system of aft additives which includes an operation corresponding to each of the unary polarized additives. This means we require operations $(\ )^*$ and $(\ )_*$ which give polarized polynatural equivalences:

$$\widehat{\mathbf{X}}(\Gamma, X, \Gamma' \,;\, \Delta) \cong \mathbf{X}_p(\Gamma \,/\, X^* \setminus \Gamma' \,;\, \Delta) \;\text{ and }\; \widehat{\mathbf{X}}(\Gamma \,;\, \Delta, X, \Delta') \cong \mathbf{X}_p(\Gamma \,;\, \Delta \,/\, X_* \setminus, \Delta)$$

see also the rules in Table 9. We shall call the system with these "focusing" operations an *inner polarized polycategory* (since these focusing operations provide an inner adjoint to the identity).

Our first observation (which is the poly-version of Lemma 3.2.1) tells us that if we wish to consider aft additives together with polarized additives in a logic it is more economical to consider an inner polarization with aft additives as this already has polarized additives:

5.5.1. Lemma. *An inner polarized polycategory with aft products and coproducts has polarized products and coproducts.*

Proof. We set $\bigsqcup_{i\in I} X_i = \bigvee_{i\in I} X_i^*$; note the following polynatural equivalences ($\bigsqcap$ is dual):

$$\mathbf{X}_p(\Gamma \,/\, \textstyle\bigvee_{i\in I} X_i^* \setminus \Gamma' \,;\, \Delta) \cong \displaystyle\prod_{i\in I} \mathbf{X}_p(\Gamma \,/\, X_i^* \setminus \Gamma' \,;\, \Delta) \cong \prod_{i\in I} \widehat{\mathbf{X}}(\Gamma, X_i, \Gamma' \,;\, \Delta)$$

∎

Inner polarized polycategories with aft additives form an evident 2-category in which 1-cells are polyfunctors which preserve the additive structure and whose 2-cells are polynatural transformations. We shall denote this 2-category by $\mathbf{iPolPolyCat}_{\vee\wedge}$. There is an obvious underlying 2-functor $U\colon \mathbf{iPolPolyCat}_{\vee\wedge} \longrightarrow \mathbf{PolPolyCat}$, which simply forgets the inner and additive structure, which has a left 2-adjoint.

5.5.2. Theorem. *There is a 2-adjunction*

$$\begin{tikzcd} \mathbf{PolPolyCat} \arrow[r, bend right, "\mathsf{Gam}_{\vee\wedge}"'] \arrow[r, phantom, "\top"] & \mathbf{iPolPolyCat}_{\vee\wedge} \arrow[l, bend right, "\mathsf{U}"'] \end{tikzcd}$$

We shall prove this theorem by providing an explicit construction of the free inner polarized polycategory with aft additives. To do this we shall develop a term calculus for the additives and the inner polarization. It turns out that, much as polarized game polycategories do, inner polarized polycategories with additives have a rather compelling reading as processes; this gives another motivation for describing the term logic in some detail.

The term logic is described schematically in Table 10. Note that we have not accounted for all the variants that polarization and focusing requires, also note, for example, that the position of a channel in the input or output determines whether a term is derived from tupling or cotupling. This allows for a high degree of economy in the notation which, admittedly, has the drawback that there is implicit typing and polarization information which is important.

The cut-elimination procedure for this term logic produces a confluent reduction system modulo equations. We may actually break the terms into three parts: those concerned purely with the additives, those concerned purely with the focusing, and those concerned with the interaction of the two systems. Notice that many of the equations essentially assert that messages sent on different channels are independent (here α is always assumed different from β).

We start with the cut eliminations steps for the additive part of the logic:

$$(\alpha[a_k]f) \;;_\beta\; g \quad\Rightarrow\quad \alpha[a_k](f \;;_\beta\; g) \qquad\qquad f \;;_\beta\; (\alpha[a_k]g) \quad\Rightarrow\quad \alpha[a_k](f \;;_\beta\; g)$$

$$(\alpha[a_k]f) \;;_\alpha\; \alpha\left\{\begin{matrix} a_1 \mapsto g_1 \\ \dots \\ a_n \mapsto g_n \end{matrix}\right\} \;\Rightarrow\; f \;;_\alpha\; g_k \qquad\qquad \alpha\left\{\begin{matrix} a_1 \mapsto g_1 \\ \dots \\ a_n \mapsto g_n \end{matrix}\right\} ;_\alpha\; (\alpha[a_k]f) \;\Rightarrow\; g_k \;;_\alpha\; f$$

$$f \;;_\alpha\; \beta\left\{\begin{matrix} b_1 \mapsto g_1 \\ \dots \\ b_n \mapsto g_n \end{matrix}\right\} \Rightarrow \beta\left\{\begin{matrix} b_1 \mapsto f \;;_\alpha\; g_1 \\ \dots \\ b_n \mapsto f \;;_\alpha\; g_n \end{matrix}\right\} \qquad \beta\left\{\begin{matrix} b_1 \mapsto g_1 \\ \dots \\ b_n \mapsto g_n \end{matrix}\right\} ;_\alpha\; h \Rightarrow \beta\left\{\begin{matrix} b_1 \mapsto g_1 \;;_\alpha\; h \\ \dots \\ b_n \mapsto g_n \;;_\alpha\; h \end{matrix}\right\}$$

These need to be combined with the following equations:

Table 10: Terms for the logic of additive inner polarized cut

$$\frac{}{\alpha{:}\, A \xrightarrow[\alpha\equiv\beta]{} \beta{:}\, A} \qquad \frac{f \in \mathbf{X}(\Gamma\,;\,\Delta)}{\Gamma \xrightarrow[f]{} \Delta}$$

$$\frac{\left\{\Gamma \,/\, \alpha{:}\, X_i \setminus \Gamma' \xrightarrow[f_i]{p} \Delta\right\}_{i\in I}}{\Gamma \,/\, \alpha{:} \bigvee_{i\in I} a_i{:}\, X_i \setminus \Gamma' \xrightarrow[\alpha\left\{\begin{array}{c} a_1 \mapsto f_1 \\ \dots \\ a_n \mapsto f_n \end{array}\right\}]{p} \Delta}\ \text{cotup} \qquad \frac{\left\{\Gamma \xrightarrow[g_i]{o} \Delta \,/\, \beta{:}\, Y_i \setminus \Delta'\right\}_{i\in I}}{\Gamma \xrightarrow[\beta\left\{\begin{array}{c} b_1 \mapsto g_1 \\ \dots \\ b_n \mapsto g_n \end{array}\right\}]{o} \Delta \,/\, \beta{:} \bigwedge_{i\in I} b_i{:}\, Y_i \setminus \Delta'}\ \text{tup}$$

$$\frac{\Gamma \xrightarrow[f]{} \Delta, \beta{:}\, Y_k, \Delta'}{\Gamma \xrightarrow[\beta[b_k]f]{} \Delta, \beta{:} \bigvee_{i\in I} b_i{:}\, Y_i, \Gamma'}\ \text{injection} \qquad \frac{\Gamma, \alpha{:}\, X_k, \Gamma' \xrightarrow[f]{} \Delta}{\Gamma, \alpha{:} \bigwedge_{i\in I} a_i{:}\, X_i, \Gamma' \xrightarrow[\alpha[a_k]f]{} \Delta}\ \text{projection}$$

$$\frac{\Gamma, \alpha{:}\, X, \Gamma' \xrightarrow[f]{op} \Delta}{\Gamma \,/\, \alpha{:}\, X^* \setminus \Gamma' \xrightarrow[\alpha\langle * \,\mapsto\, f\rangle]{p} \Delta}\ \text{focus}_L \qquad \frac{\Gamma, \beta{:}\, X, \Gamma' \xrightarrow[f]{op} \Delta}{\Gamma \xrightarrow[\alpha\langle * \,\mapsto\, f\rangle]{o} \Delta \,/\, \beta{:}\, X_* \setminus \Delta'}\ \text{focus}_R$$

$$\frac{\Gamma \xrightarrow[f]{o} \Delta \,/\, \beta{:}\, X \setminus \Delta'}{\Gamma \xrightarrow[\alpha\langle * \rangle f]{op} \Delta, \beta{:}\, X^*, \Delta'}\ \text{unfocus}_R \qquad \frac{\Gamma \,/\, \alpha{:}\, X \setminus \Gamma' \xrightarrow[f]{p} \Delta}{\Gamma, \alpha{:}\, X_*, \Gamma' \xrightarrow[\alpha\langle * \rangle f]{op} \Delta}\ \text{unfocus}_L$$

$$\frac{\Gamma \xrightarrow{f} \Delta, \alpha{:}\, X \qquad \alpha{:}\, X, \Gamma' \xrightarrow{g} \Delta'}{\Gamma, \Gamma' \xrightarrow[f;_\alpha g]{} \Delta\Delta'}\ \text{cut}$$

$$\alpha[a_i]\beta[b_j]f = \beta[b_j]\alpha[a_i]f \qquad \beta[b_j]\alpha\left\{\begin{array}{c} a_1 \mapsto f_1 \\ \dots \\ a_n \mapsto f_n \end{array}\right\} \stackrel{\Rightarrow}{\equiv} \alpha\left\{\begin{array}{c} a_1 \mapsto \beta[b_j]f_1 \\ \dots \\ a_n \mapsto \beta[b_j]f_n \end{array}\right\}$$

Notice that it is possible to orient the second of these equation (left to right) although the logic does not mandate this. When we orient this equation, we shall refer to the normal form that results (which is determined up to the first equality) as the "need-to-know" form. The motivation for considering this is discussed below.

This system stands on its own as a term logic for polarized polycategories with additives and it is not hard to see that it is confluent modulo the equations.

Next we consider the focusing terms:

$$\begin{array}{rclrcl} \alpha\langle * \rangle f \;;_\alpha\; \alpha\langle * \mapsto g\rangle & \Rightarrow & f \;;_\alpha\; g & \alpha\langle * \mapsto f\rangle \;;_\alpha\; \alpha\langle * \rangle g & \Rightarrow & f \;;_\alpha\; g \\ (\alpha[*]f) \;;_\beta\; g & \Rightarrow & \alpha[*](f \;;_\beta\; g) & g \;;_\beta\; (\alpha[*]f) & \Rightarrow & \alpha[*](g \;;_\beta\; f) \\ \alpha\langle * \mapsto f\rangle \;;_\beta\; g & \Rightarrow & \alpha\langle * \mapsto f \;;_\beta\; g\rangle & g \;;_\beta\; \alpha\langle * \mapsto f\rangle & \Rightarrow & \alpha\langle * \mapsto g \;;_\beta\; f\rangle \end{array}$$

These also stand on their own as a confluent system (with no equations) for inner polarized polycategories. This is actually immediate as it is just the system for polarized unary additives which we saw with slightly different notation earlier.

Finally the interaction between the focusing terms and the additive terms is given by:

$$\beta[b_j]\alpha\langle * \rangle f = \alpha\langle * \rangle\beta[b_j]f \qquad \beta[b_j]\alpha\langle * \mapsto f\rangle \stackrel{\Rightarrow}{\equiv} \alpha\langle * \mapsto \beta[b_j]f\rangle$$

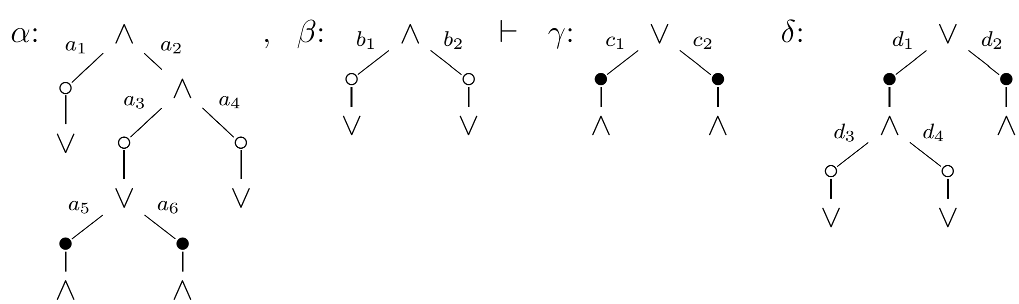
Again the second equation can be oriented (left to right) to obtain the "need-to-know" form of the term. It is worth noting that a tuple or cotuple term can only occur if that channel is in focus. If the channel is not in focus one must have performed an unfocusing step first to obtain a mixed process. If you are reading the terms in a proof search direction then it is tempting to call this focusing and, indeed, we shall use this word ambiguously for terms.

This whole system now forms a confluent system modulo the equations and the evaluation in the polarized polycategory on which it is built.

5.5.3. Proposition. *The above term logic is confluent modulo the equations and evaluation in* $\mathbf{X}$ *and provides the free inner polarized polycategory with aft additives, that is, the free polarized game polycategory,* $\mathsf{Gam}_{\vee\wedge}(\mathbf{X})$.

The details of the proof of this are lengthy but standard: for similar systems see [P03]. From this observation it is easy to obtain the proof of the theorem.

In order to illustrate this system consider the following sequent in the initial inner polarized category with aft additives. We have represented the formulas as trees in which the black and white bullets indicate the focusing constructors and the coproducts are represented with a $\vee$ and the products with a $\wedge$:



The following is a term which describes a proof of this sequent for which we shall give a process interpretation:

$$\alpha[a_1]\alpha[a_3]\beta[b_3]\alpha\langle *\rangle\delta[d_1]\alpha\left\{\begin{array}{l} a_5 \mapsto \alpha\left\langle * \mapsto \gamma[c_1]\delta[*]\delta\left\{\begin{array}{l} d_3 \mapsto \delta\langle * \mapsto \gamma[*]\gamma()\rangle \\ d_4 \mapsto \delta\langle * \mapsto \gamma[*]\gamma()\rangle \end{array}\right\}\right\rangle \\ a_6 \mapsto \alpha\left\langle * \mapsto \gamma[c_2]\delta\langle *\rangle\delta\left\{\begin{array}{l} d_3 \mapsto \delta\left\langle * \mapsto \gamma\langle *\rangle\gamma()\right\rangle \\ d_4 \mapsto \delta\left\langle * \mapsto \gamma\langle *\rangle\gamma()\right\rangle \end{array}\right\}\right\rangle \end{array}\right\}$$

The process can be explained as Mike sitting in his office, as before, with a number of telephones connected to other offices. He has two white telephones, α is to Mary's office and β to Jane's office, and two black phones, γ to Charles's office and δ to Doug's office. Mike starts in a mixed state: this time this means he is allowed to send messages to any of his colleagues (he does so to Mary, and Jane). He then tells Mary that he is expecting a response from her, $\alpha\langle *\rangle$. Mike is now listening for Mary's response and is not allowed to send messages to her. However, this does not stop him from sending other messages out and, in fact, he does send a message d_1 to Doug. (Don't you just hate people who do this on the phone?!) Then he gets Mary's response back: let us suppose it is a_6. Mike must continue to listen to Mary until she indicates that she has finished speaking which she does with $\alpha\langle * \mapsto \ldots\rangle$ to indicate she is back in listening mode.

One may think of the time between when Mike says he is listening to Mary and when she says she has finished speaking as a locking mechanism: in that period he is not allowed to send messages to her. We noted, however, that during this period he *can* send messages to other people. Now one may think that this could cause a problem because, for example, Mary can also send messages to other people in that period. Perhaps someone might receive a message from both Mike and Mary and not know the intended order of the messages. It turns out that this cannot happen: the point to remember is that Mary and Mike talk (outside to themselves) to a disjoint set of people so this clash can never happen.

When Mary releases Mike from having to listen to her, he sends Charles a c_2 and tells Doug that he is expecting a response from him. Doug responds with a d_3 say and releases Mike who now tells Charles he now is expecting a response him. Charles, however, finishes the process so that Mike is left listening to him.

Mike could have taken a more secretive approach in these communications: he could never have volunteered information that was not strictly needed to ensure that the process continued. This results in the following equivalent "need-to-know" normalized form for the interaction:

$$\alpha[a_1]\alpha[a_3]\alpha[*]\alpha\left\{\begin{array}{l} a_5 \mapsto \alpha\left\langle * \;\mapsto\; \delta[d_1]\delta[*]\delta\left\{\begin{array}{l} d_3 \mapsto \delta\langle * \mapsto \gamma[c_1]\cdot\gamma\langle *\rangle\gamma()\rangle \\ d_4 \mapsto \delta\langle * \mapsto \gamma[c_1]\cdot\gamma\langle *\rangle\gamma()\rangle \end{array}\right\}\right\rangle \\ a_6 \mapsto \alpha\left\langle * \mapsto \delta[d_1]\delta\langle *\rangle\delta\left\{\begin{array}{l} d_3 \mapsto \delta\langle * \mapsto \gamma[c_2]\gamma\langle *\rangle\gamma()\rangle \\ d_4 \mapsto \delta\langle * \mapsto \gamma[c_2]\gamma\langle *\rangle\gamma()\rangle \end{array}\right\}\right\rangle \end{array}\right\}$$

Delaying a communication until it is absolutely necessary has the effect of collecting all the communications along a particular channel to the point just before we listen on that channel or hit a "component". When there are no components the effect of delaying will be to remove any communications which are unnecessary as they get pushed onto an empty tuple. It is easy to see that the result of working in a "need-to-know" fashion makes the communication exactly the same as that supported by the polarized additives. Thus, the initial inner polarized polycategory with additives is actually equivalent to the initial polarized game polycategory.

5.5.4. Corollary. $\mathsf{Gam}(\mathbf{0})$ *(combinatorial AJ games), the free polarized game polycategory on the initial module, is equivalent to* $\mathsf{Gam}_{\vee\wedge}(\mathbf{0})$.

This may be seen directly from the above discussion using the "need-to-know" form of the process. However, it may also be seen categorically since $\mathsf{Gam}(\mathbf{0})$ already has additives: every object in the player category, for example, is already a polarized coproduct and coproducts of these always exist.

5.6. Adding fore additives freely. One may always add free products or coproducts to a category using the "family" construction. If we start with a polarized game category $\mathbf{X} = \langle \mathbf{X}_o, \mathbf{X}_p, \widehat{\mathbf{X}} \rangle$, we may freely add fore additives by adding coproducts freely to $\mathbf{X}_o$ and products freely to $\mathbf{X}_p$, to obtain $\mathrm{Fam}(\mathbf{X}) = \langle \Sigma\mathbf{X}_o, \Pi\mathbf{X}_p, \mathrm{Fam}(\widehat{\mathbf{X}}) \rangle$. Objects of $\Sigma\mathbf{X}_o$ are I-indexed families $(X_i)_{i\in I}$ of objects of $\mathbf{X}_o$ (for I a finite set); we may denote such families by formal sums $\sum_{i\in I} X_i$. A morphism $f\colon \sum_{i\in I} X_i \longrightarrow \sum_{j\in J} X'_j$ is given by a function $f\colon I \longrightarrow J$ and a family $(f_i\colon X_i \longrightarrow X'_{f(i)})_{i\in I}$ of morphisms of $\mathbf{X}_o$. Dually, objects of $\Pi\mathbf{X}_p$ are I-indexed families of objects of $\mathbf{X}_p$, denoted by formal products $\prod_{i\in I} Y_i$, and a morphism $g\colon \prod_{i\in I} Y_i \longrightarrow \prod_{j\in J} Y'_j$ is given by a function $g\colon J \longrightarrow I$ and a J-indexed family $(g_j\colon Y_{g(j)} \longrightarrow Y'_j)_{j\in J}$. Finally, a module map $\sum_{i\in I} X_i \longrightarrow \prod_{j\in J} Y_j$ is an $I \times J$ matrix of maps $(f_{ij}\colon X_i \longrightarrow\!\!\!\!\!+\ Y_j)_{i,j\in I\times J}$ with the obvious definitions for composition. In $\mathrm{Fam}(\mathbf{X})$, $\Sigma\mathbf{X}_o$ has sums represented by the indexed families, and $\Pi\mathbf{X}_p$ has products represented by its indexed families, which justifies the notation using formal sums and products.

But although $\mathrm{Fam}(\mathbf{X})$ is a polarized category, it is not necessarily a polarized game category: we would hope that if $\mathbf{X}$ were a polarized game category, we could lift the game structure to $\mathrm{Fam}(\mathbf{X})$. In particular, we want to lift an adjunction of the form

$$\begin{tikzcd} \mathbf{X}_o \arrow[r, bend right, "\sqcup_1"'] \arrow[r, phantom, "\top"] & \mathbf{X}_p \arrow[l, bend right, "\sqcap_1"'] \end{tikzcd}$$

to the family construction.

$$\begin{tikzcd} \Sigma\mathbf{X}_o \arrow[r, bend right, "\sqcup_1"'] \arrow[r, phantom, "\top"] & \Pi\mathbf{X}_p \arrow[l, bend right, "\sqcap_1"'] \end{tikzcd}$$

As usual, we shall denote $\bigsqcup_1 Q = Q^*$ and $\bigsqcap_1 P = P_*$. Then the extension of the adjunction to $\mathrm{Fam}(\mathbf{X})$ is given as follows: for a family $\sum_{i\in I} X_i$ in $\Sigma\mathbf{X}_o$, $(\sum_{i\in I} X_i)^*$ is the singleton family $(\bigsqcup_{i\in I} X_i)$. Dually, for $\prod_{i\in I} Y_i$ in $\Pi\mathbf{X}_p$, $(\prod_{i\in I} Y_i)_*$ is the singleton family $(\bigsqcap_{i\in I} Y_i)$.

It is now easy to verify that $\mathrm{Fam}(\mathbf{X})$ has polarized products and sums.

$$\begin{array}{c}
\bigsqcup_I X_i \longrightarrow (Y_j)_J \\
\hline\hline
\{X_i \longrightarrow\!\!\!\!\!+\ (Y_j)_J\}_I \\
\hline\hline
\{X_i \longrightarrow\!\!\!\!\!+\ Y_j\}_{I\times J} \\
\hline\hline
\vdots \\
\hline\hline
(X_i)_I \longrightarrow \bigsqcap_J Y_j
\end{array}$$

Consider the 2-category of polarized game categories with polarized and fore additives, and functors which preserve fore additive structure and the lifting adjunction. Because of the properties of adjoints this is equivalent to preserving both the fore and polarized (*i.e.* all the) additive structure. If we denote this by $\mathbf{PolGam}_{+\times}$, we then have an adjunction between this and our polarized game categories.

5.6.1. PROPOSITION. *There is an adjunction*

$$\begin{tikzcd} \mathbf{PolGam} \arrow[r, bend right, "\mathsf{Fam}"'] \arrow[r, phantom, "\top"] & \mathbf{PolGam}_{+\times} \arrow[l, bend right, "U"'] \end{tikzcd}$$

PROOF. We need to establish a bijection

$$\frac{\mathrm{Fam}(\mathbf{X}) \longrightarrow \mathbf{Y} \ \text{ in } \ \mathbf{PolGam}_{+\times}}{\mathbf{X} \longrightarrow U(\mathbf{Y}) \ \text{ in } \mathbf{PolGam}}$$

But a polarized fore additive functor $\mathrm{Fam}(\mathbf{X}) \longrightarrow \mathbf{Y}$ is completely determined by a polarized game functor $\mathbf{X} \longrightarrow U(\mathbf{Y})$ in our sense from the property of the family construction and the fact that polarized sums and products are preserved in both settings. Conversely given $\mathbf{X} \longrightarrow U(\mathbf{Y})$ in **PolGam** we may freely and uniquely extend it to $\mathrm{Fam}(\mathbf{X}) \longrightarrow \mathbf{Y}$ while preserving the fore additive structure (and the polarized additive structure) to obtain a fore additive functor. ■

The construction and proposition above extends to the polarized polycategory context. If $\mathbf{X}$ is a polarized game polycategory, then $\mathrm{Fam}(\mathbf{X}) = \langle \Sigma\mathbf{X}_o, \Pi\mathbf{X}_p, \mathrm{Fam}(\widehat{\mathbf{X}})\rangle$ has as O sequents (in $\Sigma\mathbf{X}_o$) collections of O sequents in $\mathbf{X}_o$ given by the evident bijective correspondences. For example, a sequent of the form $(X_i)_I \longrightarrow \ /(X'_j)_J \setminus$ would (by the freeness of the Fam construction) have to be a collection $\{X_i \longrightarrow \ / X'_{j_i} \setminus \ \}_{i\in I}$ by the following correspondences:

$$\frac{\frac{(X_i)_I \longrightarrow \ /(X'_j)_J \setminus}{\{X_i \longrightarrow \ /(X'_j)_J \setminus \ \}_{i\in I}}}{\{X_i \longrightarrow \ / X'_{j_i} \setminus \ \}_{i\in I}}$$

With sequents $\Gamma, (X_i)_I, \Gamma' \longrightarrow \Delta \, /(X'_j)_J \setminus \Delta'$ we "expand" the $\Gamma, \Gamma', \Delta, \Delta'$ just as we did the $(X_i)_I$ (keeping in mind that a family in $\mathbf{X}_o$ is treated as a sum, and one in $\mathbf{X}_p$ is treated as a product), before finally selecting the single X_j's to be "injected". P sequents are handled dually, and mixed sequents (in $\mathrm{Fam}(\widehat{\mathbf{X}})$) are just matrices. Then Proposition 5.6.1 is valid for polarized game polycategories (relative to fore additives in the obvious way). We shall denote the fore additive version of **PolPolyGam** by $\mathbf{PolPolyGam}_{+\times}$.

5.6.2. PROPOSITION. *There is an adjunction*

$$\begin{tikzcd} \mathbf{PolPolyGam} \arrow[r, bend right, "\mathsf{Fam}"'] \arrow[r, phantom, "\top"] & \mathbf{PolPolyGam}_{+\times} \arrow[l, bend right, "U"'] \end{tikzcd}$$

Part III Representability

6. Linear polarized categories

It is well known that AJ games (*i.e.* finitary Abramsky-Jagadeesan games) carry both a multiplicative and an exponential structure. In fact, the multiplicative structure was the main interest of the original paper [AJ92]. We shall explore an exponential structure, due to Lamarche and to Curien [C93] in the next section. In this section our aim is to derive the polarized multiplicative structure.

We shall approach the multiplicatives through the notion of representability in Hermida's sense, [H00], but modified for the polarized polycategorical setting. We shall explain how this gives rise to a polarized categorical doctrine and we shall provide a recipe for generating the necessary coherence diagrams (inevitably there are many).

It is possible to provide proof circuits for this polarized multiplicative structure and it is interesting to compare their form to the proof circuits for linearly distributive categories, *-autonomous categories, or MLL. As before (see sections 4.3, 5.2) one can "polarize" multiplicative circuits; it is worth noting that, in particular, all the multiplicative equivalences can be polarized. This has the (discouraging) consequence that from the point of view of multiplicative complexity there is no simplifying advantage gained by polarizing. Of course, we have already seen that there definitely is a simplifying advantage when one considers (polarized) additives.

6.1. The sequent rules for multiplicatives. We shall start by introducing the sequent rules for the polarized multiplicatives. Notice that the logic of polarized cut has two sorts of comma (on the left and on the right) and two sorts of slash–backslash to represent. The commas on the left of the turnstile are represented by a "tensor", written $\otimes$, while the commas on the right are (co)represented by a "par", which we write as $\oplus$. This implies that $\otimes$ is an operation on O types, and $\oplus$ is an operation on P types. Although this notation suggests linear logic,[4] the reader is reminded that a model of the logic is not necessarily a model of linear logic. The sequent rules for tensor are presented in Table 11; the rules for the par are dual.

In addition to the tensors there are also the tensor context binders, Table 12, which represent the context slashes (on the left) just as tensors represent commas, and so they act like tensors apart from being differently typed. There are two: $Q \mathbin{\otimes\!\rangle} P$ which binds an opponent type to a player type and produces a player type, and the other $P \mathbin{\langle\!\otimes} Q$ which binds a player type to an opponent type to produce a player type. In addition, there are two dual par context binders, which represent context slashes on the right, and so behave like par's apart from their typing: $Q \mathbin{\langle\!\oplus} P$ which produces an opponent type from an opponent type and a player type, and $P \mathbin{\oplus\!\rangle} Q$ which produces an opponent type from a player type and an opponent type. We shall illustrate the rules for $\otimes\!\rangle$ and $\langle\!\otimes$ — the others are dual. Again, note the underlying simplicity behind the apparent complications: these are the usual left

[4] Note that we are using the symbol $\oplus$ for the multiplicative disjunction "par", rather than $\parr$ which is more standard in the linear logic community (where $\oplus$ is used to denote the coproduct). This follows notation in our previous papers, and derives from the categorical roots of the subject.

Table 11: Tensor rules

$$\frac{\Gamma_1, Q_1, Q_2, \Gamma_2 \vdash_o \Delta_1 \,/\, Q \,\backslash\, \Delta_2}{\Gamma_1, Q_1 \otimes Q_2, \Gamma_2 \vdash_o \Delta_1 \,/\, Q \,\backslash\, \Delta_2}\ \otimes L_o \qquad \frac{\Gamma, Q_1, Q_2, \Gamma_2 \underset{o\ \, p}{\longmapsto} \Delta}{\Gamma_1, Q_1 \otimes Q_2, \Gamma_2 \underset{o\ \, p}{\longmapsto} \Delta}\ \otimes L_c$$

$$\frac{\Gamma_1, Q_1, Q_2, \Gamma_2 \,/\, P \,\backslash\, \Gamma \vdash_p \Delta}{\Gamma_1, Q_1 \otimes Q_2, \Gamma_2 \,/\, P \,\backslash\, \Gamma \vdash_p \Delta}\ \otimes L_{pl} \qquad \frac{\Gamma \,/\, P \,\backslash\, \Gamma_1, Q_1, Q_2, \Gamma_2 \vdash_p \Delta}{\Gamma \,/\, P \,\backslash\, \Gamma_1, Q_1 \otimes Q_2, \Gamma_2 \vdash_p \Delta}\ \otimes L_{pr}$$

$$\frac{\Gamma_1 \vdash_o \Delta_1 \,/\, Q_1 \,\backslash \qquad \Gamma_2 \vdash_o \ \,/\, Q_2 \,\backslash\, \Delta_2'}{\Gamma_1, \Gamma_2 \vdash_o \Delta_1 \,/\, Q_1 \otimes Q_2 \,\backslash\, \Delta_2'}\ \otimes R_o$$

$$\frac{\Gamma_1 \vdash_o \Delta_1 \,/\, Q_1 \,\backslash\, \Delta_1' \qquad \vdash_o \ \,/\, Q_2 \,\backslash\, \Delta_2'}{\Gamma_1 \vdash_o \Delta_1 \,/\, Q_1 \otimes Q_2 \,\backslash\, \Delta_2', \Delta_1'}\ \otimes R_{o1}$$

$$\frac{\vdash_o \Delta_1 \,/\, Q_1 \,\backslash \qquad \Gamma_2 \vdash_o \Delta_2 \,/\, Q_2 \,\backslash\, \Delta_2'}{\Gamma_2 \vdash_o \Delta_2, \Delta_1 \,/\, Q_1 \otimes Q_2 \,\backslash\, \Delta_2'}\ \otimes R_{o2}$$

$$\frac{\Gamma_1, \Gamma_2 \vdash_o \Delta_1 \,/\, Q \,\backslash\, \Delta_2}{\Gamma_1, \top, \Gamma_2 \vdash_o \Delta_1 \,/\, Q \,\backslash\, \Delta_2}\ \top L_o \qquad \frac{\Gamma_1, \Gamma_2 \underset{o\ \, p}{\longmapsto} \Delta}{\Gamma_1, \top, \Gamma_2 \underset{o\ \, p}{\longmapsto} \Delta}\ \top L_c$$

$$\frac{\Gamma_1, \Gamma_2 \,/\, P \,\backslash\, \Gamma \vdash_p \Delta}{\Gamma_1, \top, \Gamma_2 \,/\, P \,\backslash\, \Gamma \vdash_p \Delta}\ \top L_{pl} \qquad \frac{\Gamma \,/\, P \,\backslash\, \Gamma_1, \Gamma_2 \vdash_p \Delta}{\Gamma \,/\, P \,\backslash\, \Gamma_1, \top, \Gamma_2 \vdash_p \Delta}\ \top L_{pr}$$

$$\frac{}{\vdash_o \ \,/\, \top \,\backslash}\ \top R_o$$

Table 12: Tensor context rules

$$\frac{\Gamma \,/\, P \,\backslash\, Q, \Gamma' \vdash_p \Delta}{\Gamma \,/\, P \mathbin{(\!\otimes} Q \,\backslash\, \Gamma' \vdash_p \Delta}\ {(\!\otimes}L_p \qquad\qquad \frac{\Gamma, Q \,/\, P \,\backslash\, \Gamma' \vdash_p \Delta}{\Gamma \,/\, Q \mathbin{\otimes\!)} P \,\backslash\, \Gamma' \vdash_p \Delta}\ {\otimes\!)}L_p$$

$$\frac{\Phi \,/\, X \,\backslash\, \Phi' \vdash_p \Psi', P, \Psi \quad \Gamma \vdash_o \ \,/\, Q \,\backslash\, \Delta}{\Phi \,/\, X \,\backslash\, \Phi', \Gamma \vdash_p \Psi', P \mathbin{(\!\otimes} Q, \Psi, \Delta}\ {(\!\otimes}R_{po} \qquad \frac{\Gamma \vdash_o \Delta \,/\, Q \,\backslash\ \quad \Phi \,/\, X \,\backslash\, \Phi' \vdash_p \Psi, P, \Psi'}{\Gamma, \Phi \,/\, X \,\backslash\, \Phi' \vdash_p \Psi, \Delta, Q \mathbin{\otimes\!)} P, \Psi'}\ {\otimes\!)}R_{op}$$

where one of Γ or Ψ is empty

$$\frac{\Phi \underset{o\;p}{\vdash\!\!\!\to} \Psi', P, \Psi \quad \Gamma \vdash_o \ \,/\, Q \,\backslash\, \Delta}{\Phi, \Gamma \underset{o\;p}{\vdash\!\!\!\to} \Psi', P \mathbin{(\!\otimes} Q, \Psi, \Delta}\ {(\!\otimes}R_{co} \qquad \frac{\Gamma \vdash_o \Delta \,/\, Q \,\backslash\ \quad \Phi \underset{o\;p}{\vdash\!\!\!\to} \Psi, P, \Psi'}{\Gamma, \Phi \underset{o\;p}{\vdash\!\!\!\to} \Psi, \Delta, Q \mathbin{\otimes\!)} P, \Psi'}\ {\otimes\!)}R_{oc}$$

where one of Γ or Ψ is empty

$$\frac{\Phi \vdash_o \Delta \,/\, Q' \,\backslash\, \Delta', P, \Theta \quad \Gamma \vdash_o \Psi \,/\, Q \,\backslash\, \Delta''}{\Phi, \Gamma \vdash_o \Psi, \Delta \,/\, Q' \,\backslash\, \Delta', P \mathbin{(\!\otimes} Q, \Delta'', \Theta}\ {(\!\otimes}R_{oo} \quad \frac{\Gamma \vdash_o \Delta \,/\, Q \,\backslash\, \Psi \quad \Phi \vdash_o \Theta, P, \Delta' \,/\, Q' \,\backslash\, \Delta''}{\Gamma, \Phi \vdash_o \Theta, \Delta, Q \mathbin{\otimes\!)} P, \Delta' \,/\, Q' \,\backslash\, \Delta'', \Psi}\ {\otimes\!)}R_{oo}$$

where both Γ, Ψ or both Ψ, Θ or both Φ, Θ are empty

and right introduction rules for tensor, treating the context slashes as if they were commas, with adaptations due to typing and planarity.

Comment: as a mnemonic, one should recall that these context binder operators point in the direction of the type whose type is inherited by the compound type.

This sequent calculus with the polarized cuts admits cut elimination. This can be most easily seen from the fact that the underlying unpolarized logic admits cut elimination and each step can be polarized. Hence:

6.1.1. Theorem. *The game logic with multiplicatives satisfies (categorical) cut elimination.*

6.2. Negation: closed polarized polycategories. We will say that a logic of polarized cuts is closed in case there are four negations. The first two $P \mapsto {}^{\perp}P$ and $P \mapsto P^{\perp}$ take player propositions to opponent propositions. The second two $Q \mapsto {}^{\perp}Q$ and $Q \mapsto Q^{\perp}$ take opponent propositions to player propositions. We use the same symbols for these two pairs of operations, since the typing makes it clear which is intended. Of course, these are the typed versions of the usual two negation operators of non-commutative linear logic, and like them they satisfy the standard inference rules for negation given in Table 13

It is worth noting that these rules are bijective (reversible). In fact, a simpler alternative equivalent presentation of the four negations may be given by the axioms of Table 14; using these and cut, it is a routine matter to verify bijective correspondences corresponding to the rules of Table 13. Note also that these negation operators are "inverse": ${}^{\perp}(X^{\perp}) \cong X \cong ({}^{\perp}X)^{\perp}$ for X either a player or opponent proposition. (This is essentially the statement that polarized linear adjoints are unique up to isomorphism.) It is also straightforward to show that ${}^{\perp}\bot = \bot^{\perp} = \top$ and *vice versa.* These facts are easily proved; doing so using the circuit terms below is a pleasant exercise.

Table 13: Negation deduction rules

$\dfrac{\Gamma / X \setminus \Gamma' \vdash_p P, \Delta}{{}^{\perp}P, \Gamma / X \setminus \Gamma' \vdash_p \Delta}\ {}^{\perp}L_p$	$\dfrac{\Gamma \vdash_o P, \Delta / X \setminus \Delta'}{{}^{\perp}P, \Gamma \vdash_o \Delta / X \setminus \Delta'}\ {}^{\perp}L_o$	$\dfrac{\Gamma' \vdash_o \ / Q \setminus \Delta'}{/ {}^{\perp}Q \setminus \Gamma' \vdash_p \Delta'}\ {}^{\perp}L_{op}$	$\dfrac{\Gamma \underset{o\,p}{\mapsto} P, \Delta}{{}^{\perp}P, \Gamma \underset{o\,p}{\mapsto} \Delta}\ {}^{\perp}L_c$
$\dfrac{\Gamma / X \setminus \Gamma' \vdash_p \Delta, P}{\Gamma / X \setminus \Gamma', P^{\perp} \vdash_p \Delta}\ L_p^{\perp}$	$\dfrac{\Gamma \vdash_o \Delta / X \setminus \Delta', P}{\Gamma, P^{\perp} \vdash_o \Delta / X \setminus \Delta'}\ L_o^{\perp}$	$\dfrac{\Gamma \vdash_o \Delta / Q \setminus}{\Gamma / Q^{\perp} \setminus \ \vdash_p \Delta}\ L_{op}^{\perp}$	$\dfrac{\Gamma \underset{o\,p}{\mapsto} \Delta, P}{\Gamma, P^{\perp} \underset{o\,p}{\mapsto} \Delta}\ L_c^{\perp}$
$\dfrac{Q, \Gamma / X \setminus \Gamma' \vdash_p \Delta}{\Gamma / X \setminus \Gamma' \vdash_p {}^{\perp}Q, \Delta}\ {}^{\perp}R_p$	$\dfrac{\Gamma, Q \vdash_o \Delta / X \setminus \Delta'}{\Gamma \vdash_o \Delta / X \setminus \Delta', Q^{\perp}}\ {}^{\perp}R_o$	$\dfrac{\Gamma / P \setminus \ \vdash_p \Delta}{\Gamma \vdash_o \Delta / {}^{\perp}P \setminus}\ {}^{\perp}R_{po}$	$\dfrac{\Gamma, Q \underset{o\,p}{\mapsto} \Delta}{\Gamma \underset{o\,p}{\mapsto} \Delta, {}^{\perp}Q}\ {}^{\perp}R_c$
$\dfrac{\Gamma / X \setminus \Gamma', Q \vdash_p \Delta}{\Gamma / X \setminus \Gamma' \vdash_p Q^{\perp}, \Delta}\ R_p^{\perp}$	$\dfrac{Q, \Gamma \vdash_o \Delta / X \setminus \Delta'}{\Gamma \vdash_o Q^{\perp}, \Delta / X \setminus \Delta'}\ R_o^{\perp}$	$\dfrac{/ P \setminus \Gamma' \vdash_p \Delta'}{\Gamma' \vdash_o \ / P^{\perp} \setminus \Delta'}\ R_{po}^{\perp}$	$\dfrac{Q, \Gamma \underset{o\,p}{\mapsto} \Delta}{\Gamma \underset{o\,p}{\mapsto} Q^{\perp}, \Delta}\ R_c^{\perp}$

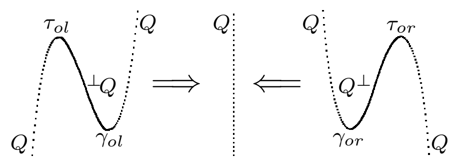
Table 14: Negation axioms

$\dfrac{}{\vdash_o \ / Q \setminus {}^{\perp}Q}\ \tau_{ol}$	$\dfrac{}{\vdash_o Q^{\perp} / Q \setminus}\ \tau_{or}$
$\dfrac{}{\vdash_o P / {}^{\perp}P \setminus}\ \tau_{pl}$	$\dfrac{}{\vdash_o \ / P^{\perp} \setminus P}\ \tau_{pr}$
$\dfrac{}{/ {}^{\perp}Q \setminus Q \vdash_p}\ \gamma_{ol}$	$\dfrac{}{Q / Q^{\perp} \setminus \ \vdash_p}\ \gamma_{or}$
$\dfrac{}{{}^{\perp}P / P \setminus \ \vdash_p}\ \gamma_{pl}$	$\dfrac{}{/ P \setminus P^{\perp} \vdash_p}\ \gamma_{pr}$

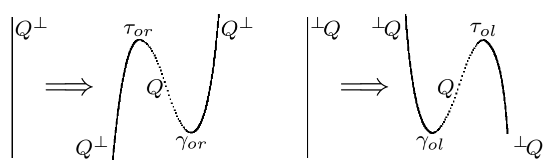
The axioms of Table 14 correspond to our previous treatment of negation in the non polarized context (see [CS97a, CKS] for example), and prompt the following definition.

6.2.1. Definition. *A polarized polycategory is* closed *in case for each* P*-object* X *there are polyarrows corresponding to the "negation axioms" of Table 14 which in addition satisfy four equations illustrated by the circuit rewrites below.*

We illustrate two, for an O-object; there are two similar ones for P-objects. The axioms τ, γ are represented by labelled bends in the wires.



In the free case these may be regarded as reductions to which one has to add four expansion rewrites in order to obtain a confluent terminating reduction/expansion system for free closed polarized polycategories based on components. We illustrate the two for O-types; there are two more similar ones for P-types.



Clearly closed polarized polycategories can be organized into a category **CPolPolyCat**, and using the construction of the free polarized polycategory we can now construct the free closed polarized polycategory on components:

6.2.2. Proposition. *The underlying functor* $U\colon \mathbf{CPolPolyCat} \longrightarrow \mathbf{PolyGraph}$ *has a left adjoint* F_c *which associates to each polarized polygraph its polarized polycategory of proof circuits with negation links subject to the confluent and terminating expansion/reduction system above.*

It is interesting to note that the analogue of Lemma 4.3.2 also holds for free closed polarized polycategories. This is because the extra circuit components required to express negation are all focused and Lemma 4.3.2 can be applied to the circuit in which the negation links are regarded as (focused) components. We therefore have:

6.2.3. Lemma. *In the free closed polarized polycategory* $F_c(\mathbf{G})$

1. *Any focused polyarrow* $g \in F_c(\mathbf{G})$ *contains no unfocussed components;*

2. *Any unfocused polyarrow* $f \in F_c(\mathbf{G})$ *must contain exactly one unfocused component.*

The game construction can be used to add polarized products and coproducts to a closed polarized polycategory $\mathbf{X}$. However, notice that it is not immediately clear that $\mathsf{Gam}(\mathbf{X})$ should be closed. In fact, it is and to see this we note that we can inductively define negations. For example, for an O-object X, we can let $(\prod_{i\in I} X_i)^\perp :=_{\mathsf{def}} \coprod_{i\in I} X_i^\perp$, constructing the required polyarrow τ_{or} from the following sequent derivation:

$$\frac{\left\{\dfrac{\vdash_o X_i^\perp \,/\, X_i \setminus}{\underset{o\ p}{\mapsto} \coprod_{i\in I} X_i^\perp, X_i}\right\}_{i\in I}}{\vdash_o \coprod_{i\in I} X_i^\perp \,/\, \prod_{i\in I} X_i \setminus}$$

Aft additives follow the same pattern: exactly the same sequent proof works but the whole proof now uses focused sequents:

$$\frac{\left\{\dfrac{\vdash_o X_i^\perp \,/\, X_i \setminus}{\vdash_o \bigvee_{i\in I} X_i^\perp \,/\, X_i \setminus}\right\}_{i\in I}}{\vdash_o \bigvee_{i\in I} X_i^\perp \,/\, \bigwedge_{i\in I} X_i \setminus}$$

Furthermore it is easily checked that the required identities hold. We therefore have:

6.2.4. Proposition.

1. *If $\mathbf{X}$ is a closed polarized polycategory then $\mathsf{Gam}(\mathbf{X})$ is closed. Thus, the 2-adjunction of Proposition 5.1.1 restricts to closed polarized polycategories.*

2. *If $\mathbf{X}$ is a closed polarized polycategory then $F_{\vee\wedge}(\mathbf{X})$ is closed. Thus, the 2-adjunction of Proposition 5.4.1 restricts to closed polarized polycategories.*

3. *If $\mathbf{X}$ is a closed polarized polycategory then $\mathsf{Gam}_{\vee\wedge}(\mathbf{X})$ is closed. Thus, the 2-adjunction of Theorem 5.5.2 restricts to closed polarized polycategories.*

Notice that the final polarized polycategory, $\mathbf{1}$, is closed. This means that the additive completion $F_{\vee\wedge}(\mathbf{1})$ is closed. The negation in this case is given by the obvious duality in the construction. Notice that $F_{\vee\wedge}(\mathbf{1})$ does not have polarized products or coproducts as it is not inner. Furthermore, notice that while $\mathsf{Gam}(\mathbf{1})$, $F_{\vee\wedge}(\mathbf{1})$ and $\mathsf{Gam}_{\vee\wedge}(\mathbf{1})$ are all closed, they are all distinct.

We may extend the example of a polarized polycategory generated by types $\mathbf{T}$ (a discrete polycategory) to include negation by adding all the negation links (on the same type) and the identity links. This is then a polarized polycategory $\mathbf{T}_\neg$ which has negation. Each of $\mathsf{Gam}(\mathbf{T}_\neg)$, $F_{\vee\wedge}(\mathbf{T}_\neg)$, and $\mathsf{Gam}_{\vee\wedge}(\mathbf{T}_\neg)$ then have negation. Note that $F_{\vee\wedge}(\mathbf{T}_\neg)$ has no mixed polyarrows.

6.3. Some example derivations. It may be useful to display some example derivations. These are mainly to illustrate what properties one can expect of models of this type theory, and to illustrate the use of the proof system.

1. $[Q_1 \mathbin{\otimes\rangle} (Q_2 \mathbin{\otimes\rangle} P) \;{}_p\!\dashv\vdash_p (Q_1 \otimes Q_2) \mathbin{\otimes\rangle} P]$

$$\dfrac{\dfrac{\dfrac{\dfrac{Q_1 \vdash_o \; /Q_1\backslash \qquad Q_2 \vdash_o \; /Q_2\backslash}{Q_1, Q_2 \vdash_o \; /Q_1 \otimes Q_2\backslash} \qquad /P\backslash \; \vdash_p P}{Q_1, Q_2 \,/P\backslash \; \vdash_p (Q_1 \otimes Q_2) \mathbin{\otimes\rangle} P}}{Q_1 \,/Q_2 \mathbin{\otimes\rangle} P\backslash \; \vdash_p (Q_1 \otimes Q_2) \mathbin{\otimes\rangle} P}}{/Q_1 \mathbin{\otimes\rangle} (Q_2 \mathbin{\otimes\rangle} P)\backslash \; \vdash_p (Q_1 \otimes Q_2) \mathbin{\otimes\rangle} P}
\qquad
\dfrac{\dfrac{\dfrac{Q_1 \vdash_o \; /Q_1\backslash \qquad \dfrac{Q_2 \vdash_o \; /Q_2\backslash \qquad /P\backslash \; \vdash_p P}{Q_2 \,/P\backslash \; \vdash_p Q_2 \mathbin{\otimes\rangle} P}}{Q_1, Q_2 \,/P\backslash \; \vdash_p Q_1 \mathbin{\otimes\rangle} (Q_2 \mathbin{\otimes\rangle} P)}}{Q_1 \otimes Q_2 \,/P\backslash \; \vdash_p Q_1 \mathbin{\otimes\rangle} (Q_2 \mathbin{\otimes\rangle} P)}}{/(Q_1 \otimes Q_2) \mathbin{\otimes\rangle} P\backslash \; \vdash_p Q_1 \mathbin{\otimes\rangle} (Q_2 \mathbin{\otimes\rangle} P)}$$

2. $[Q \mathbin{\otimes\rangle} (P_1 \vee P_2) \vdash_p Q \mathbin{\otimes\rangle} P_1 \vee Q \mathbin{\otimes\rangle} P_2]$

$$\dfrac{\dfrac{\dfrac{\dfrac{Q \vdash_o \; /Q\backslash \qquad /P_1\backslash \; \vdash_p P_1}{Q \,/P_1\backslash \; \vdash_p Q \mathbin{\otimes\rangle} P_1}}{Q \,/P_1\backslash \; \vdash_p Q \mathbin{\otimes\rangle} P_1 \vee Q \mathbin{\otimes\rangle} P_2} \qquad \dfrac{\dfrac{Q \vdash_o \; /Q\backslash \qquad /P_2\backslash \; \vdash_p P_2}{Q \,/P_2\backslash \; \vdash_p Q \mathbin{\otimes\rangle} P_2}}{Q \,/P_2\backslash \; \vdash_p Q \mathbin{\otimes\rangle} P_1 \vee Q \mathbin{\otimes\rangle} P_2}}{Q \,/P_1 \vee P_2\backslash \; \vdash_p Q \mathbin{\otimes\rangle} P_1 \vee Q \mathbin{\otimes\rangle} P_2}}{/Q \mathbin{\otimes\rangle} (P_1 \vee P_2)\backslash \; \vdash_p Q \mathbin{\otimes\rangle} P_1 \vee Q \mathbin{\otimes\rangle} P_2}$$

3. $[Q \mathbin{\otimes\rangle} P_1 \vee Q \mathbin{\otimes\rangle} P_2 \vdash_p Q \mathbin{\otimes\rangle} (P_1 \vee P_2)]$

$$\dfrac{\dfrac{\dfrac{Q \vdash_o \; /Q\backslash \qquad \dfrac{/P_1\backslash \; \vdash_p P_1}{/P_1\backslash \; \vdash_p P_1 \vee P_2}}{Q \,/P_1\backslash \; \vdash_p Q \mathbin{\otimes\rangle} (P_1 \vee P_2)}}{/Q \mathbin{\otimes\rangle} P_1\backslash \; \vdash_p Q \mathbin{\otimes\rangle} (P_1 \vee P_2)} \qquad \dfrac{\dfrac{Q \vdash_o \; /Q\backslash \qquad \dfrac{/P_2\backslash \; \vdash_p P_2}{/P_2\backslash \; \vdash_p P_1 \vee P_2}}{Q \,/P_2\backslash \; \vdash_p Q \mathbin{\otimes\rangle} (P_1 \vee P_2)}}{/Q \mathbin{\otimes\rangle} P_2\backslash \; \vdash_p Q \mathbin{\otimes\rangle} (P_1 \vee P_2)}}{/Q \mathbin{\otimes\rangle} P_1 \vee Q \mathbin{\otimes\rangle} P_2\backslash \; \vdash_p Q \mathbin{\otimes\rangle} (P_1 \vee P_2)}$$

4. $[Q_1 \otimes (Q_2 \mathbin{\langle\oplus} P_3 \wedge P_2 \mathbin{\oplus\rangle} Q_3) \vdash_o (Q_1 \otimes Q_2) \mathbin{\langle\oplus} P_3 \wedge (Q_1 \mathbin{\otimes\rangle} P_2 \vee P_1 \mathbin{\langle\otimes} Q_2) \mathbin{\oplus\rangle} Q_3]$

$$\dfrac{\dfrac{\dfrac{Q_1, Q_2 \vdash_o \; /Q_1 \otimes Q_2\backslash \qquad /P_3\backslash \; \vdash_p P_3}{Q_1, Q_2 \mathbin{\langle\oplus} P_3 \vdash_o \; /Q_1 \otimes Q_2\backslash P_3}}{Q_1, (Q_2 \mathbin{\langle\oplus} P_3 \wedge P_2 \mathbin{\oplus\rangle} Q_3) \vdash_o \; /(Q_1 \otimes Q_2) \mathbin{\langle\oplus} P_3\backslash} \qquad \dfrac{\dfrac{Q_3 \vdash_o \; /Q_3\backslash \qquad Q_1 \,/P_2\backslash \; \vdash_p Q_1 \mathbin{\otimes\rangle} P_2}{Q_1, P_2 \mathbin{\oplus\rangle} Q_3 \vdash_o Q_1 \mathbin{\otimes\rangle} P_2 \vee P_1 \mathbin{\langle\otimes} Q_2 \,/Q_3\backslash}}{Q_1, (Q_2 \mathbin{\langle\oplus} P_3 \wedge P_2 \mathbin{\oplus\rangle} Q_3) \vdash_o \; /(Q_1 \mathbin{\otimes\rangle} P_2 \vee P_1 \mathbin{\langle\otimes} Q_2) \mathbin{\oplus\rangle} Q_3\backslash}}{Q_1 \otimes (Q_2 \mathbin{\langle\oplus} P_3 \wedge P_2 \mathbin{\oplus\rangle} Q_3) \vdash_o \; /(Q_1 \otimes Q_2) \mathbin{\langle\oplus} P_3 \wedge (Q_1 \mathbin{\otimes\rangle} P_2 \vee P_1 \mathbin{\langle\otimes} Q_2) \mathbin{\oplus\rangle} Q_3\backslash}$$

where some steps are omitted.

5. $[{}^\perp P \sqcup {}^\perp P' \;{}_p\!\dashv\vdash_p {}^\perp(P \sqcap P')]$

$$\dfrac{\dfrac{\dfrac{{}^\perp P \,/P\backslash \; \vdash_p}{{}^\perp P, P \sqcap P' \underset{o\;p}{\longmapsto}}}{{}^\perp P \underset{o\;p}{\longmapsto} {}^\perp(P \sqcap P')} \qquad \dfrac{\dfrac{{}^\perp P' \,/P'\backslash \; \vdash_p}{{}^\perp P', P \sqcap P' \underset{o\;p}{\longmapsto}}}{{}^\perp P' \underset{o\;p}{\longmapsto} {}^\perp(P \sqcap P')}}{/{}^\perp P \sqcup {}^\perp P'\backslash \; \vdash_p {}^\perp(P \sqcap P')}
\qquad
\dfrac{\dfrac{\dfrac{\vdash_o P \,/{}^\perp P\backslash}{\underset{o\;p}{\longmapsto} P, {}^\perp P \sqcup {}^\perp P'} \qquad \dfrac{\vdash_o P' \,/{}^\perp P'\backslash}{\underset{o\;p}{\longmapsto} P', {}^\perp P \sqcup {}^\perp P'}}{\vdash_o \; /P \sqcap P'\backslash\, {}^\perp P \sqcup {}^\perp P'}}{/{}^\perp(P \sqcap P')\backslash \; \vdash_p {}^\perp P \sqcup {}^\perp P'}$$

6. $[Q'^\perp \oplus Q^\perp \;{}_p\!\dashv\vdash_p (Q \otimes Q')^\perp]$

$$\dfrac{\dfrac{\dfrac{Q' \,/Q'^\perp\backslash \; \vdash_p \qquad Q \,/Q^\perp\backslash \; \vdash_p}{Q, Q' \,/Q'^\perp \oplus Q^\perp\backslash \; \vdash_p}}{Q \otimes Q' \,/Q'^\perp \oplus Q^\perp\backslash \; \vdash_p}}{/Q'^\perp \oplus Q^\perp\backslash \; \vdash_p (Q \otimes Q')^\perp}
\qquad
\dfrac{\dfrac{\dfrac{\vdash_o Q^\perp \,/Q\backslash \qquad \vdash_o Q'^\perp \,/Q'\backslash}{\vdash_o Q'^\perp, Q^\perp \,/Q \otimes Q'\backslash}}{/(Q \otimes Q')^\perp\backslash \; \vdash_p Q'^\perp, Q^\perp}}{/(Q \otimes Q')^\perp\backslash \; \vdash_p Q'^\perp \oplus Q^\perp}$$

7. $[\ \vdash_o (Q \oplus {}^{\perp}Q) \wedge (P^{\perp} \oplus P)]$

$$\dfrac{\dfrac{\vdash_o\ /Q\backslash\, {}^{\perp}Q}{\vdash_o\ /Q \oplus {}^{\perp}Q\backslash} \qquad \dfrac{\vdash_o\ /P^{\perp}\backslash\, P}{\vdash_o\ /P^{\perp} \oplus P\backslash}}{\vdash_o\ /(Q \oplus {}^{\perp}Q) \wedge (P^{\perp} \oplus P)\backslash}$$

8. $[Q \otimes (Q' \sqcup Q'')\ {}_p\dashv\vdash_p (Q \otimes Q') \sqcup (Q \otimes Q'')]$

$$\dfrac{\dfrac{\dfrac{\dfrac{Q \vdash_o\ /Q\backslash \quad Q' \vdash_o\ /Q'\backslash}{Q, Q' \vdash_o\ /Q \otimes Q'\backslash}}{Q, Q' \underset{o\ p}{\longmapsto} (Q \otimes Q') \sqcup (Q \otimes Q'')} \quad \dfrac{\dfrac{Q \vdash_o\ /Q\backslash \quad Q'' \vdash_o\ /Q''\backslash}{Q, Q'' \vdash_o\ /Q \otimes Q''\backslash}}{Q, Q'' \underset{o\ p}{\longmapsto} (Q \otimes Q') \sqcup (Q \otimes Q'')}}{Q\,/Q' \sqcup Q''\backslash\ \vdash_p (Q \otimes Q') \sqcup (Q \otimes Q'')}}{/Q \otimes (Q' \sqcup Q'')\backslash\ \vdash_p (Q \otimes Q') \sqcup (Q \otimes Q'')}$$

$$\dfrac{\dfrac{\dfrac{Q \vdash_o\ /Q\backslash \quad \dfrac{Q' \vdash_o\ /Q'\backslash}{Q' \underset{o\ p}{\longmapsto} Q' \sqcup Q''}}{Q, Q' \underset{o\ p}{\longmapsto} Q \otimes (Q' \sqcup Q'')}}{Q \otimes Q' \underset{o\ p}{\longmapsto} Q \otimes (Q' \sqcup Q'')} \quad \dfrac{\dfrac{Q \vdash_o\ /Q\backslash \quad \dfrac{Q'' \vdash_o\ /Q''\backslash}{Q'' \underset{o\ p}{\longmapsto} Q' \sqcup Q''}}{Q, Q'' \underset{o\ p}{\longmapsto} Q \otimes (Q' \sqcup Q'')}}{Q \otimes Q'' \underset{o\ p}{\longmapsto} Q \otimes (Q' \sqcup Q'')}}{/(Q \otimes Q') \sqcup (Q \otimes Q'')\backslash\ \vdash_p Q \otimes (Q' \sqcup Q'')}$$

9. $[Q \otimes (P \sqcap P') \vdash_o (Q \otimes P) \sqcap (Q \otimes P')]$

$$\dfrac{\dfrac{Q \vdash_o\ /Q\backslash \quad \dfrac{|\ P \vdash_p P}{P \sqcap P' \underset{o\ p}{\longmapsto} P}}{Q, P \sqcap P' \underset{o\ p}{\longmapsto} Q \otimes P} \quad \dfrac{Q \vdash_o\ /Q\backslash \quad \dfrac{/P'\backslash\ \vdash_p P'}{P \sqcap P' \underset{o\ p}{\longmapsto} P'}}{Q, P \sqcap P' \underset{o\ p}{\longmapsto} Q \otimes P'}}{Q \otimes (P \sqcap P') \vdash_o\ /(Q \otimes P) \sqcap (Q \otimes P')\backslash}$$

This derivation is not reversible, unless we add softness. Note in the combinatorial games this is an isomorphism.

Note: as an indication of the type of coherence condition required of this, we point out that there is another (equivalent) derivation of the $Q, P \sqcap P' \underset{o\ p}{\longmapsto} Q \otimes P$ part of the above, *viz.* the following.

$$\dfrac{\dfrac{Q \vdash_o\ /Q\backslash \qquad /P\backslash\ \vdash_p P}{Q\,/P\backslash\ \vdash_p Q \otimes P}}{Q, P \sqcap P' \underset{o\ p}{\longmapsto} Q \otimes P}$$

6.3.1. Remark. *A non-distributivity* Here is an example that shows that $O \otimes (O' \wedge O'')$ is not isomorphic to $(O \otimes O') \wedge (O \otimes O'')$: $(\{2\colon(\ )\}) \otimes (\{\ \}) = (\{\ \}, \{2\colon(\{\ \})\})$ whereas $(\{2\colon(\ )\}) \otimes (2\colon\{\ \}) = (2\colon\{\ \}, \{2\colon(2\colon\{\ \})\})$. Note that $(\{\ \}) \wedge (\{\ \}) = (2\colon\{\ \})$

6.4. Representing the polarized multiplicative types. We now turn our attention to the polarized categorical proof theory of the multiplicatives. In a sense we have already done all the hard work to obtain the proof theory as we can use the notion of representability in polarized polycategories to extract the proof theory of the multiplicatives. We wish to start by reminding the reader of this technique and by describing how it may be applied in this situation. The basic ideas follow those of Hermida [H00] and are described in [CKS] for the multiplicatives of linear logic.

We shall say an object $A \otimes B$ represents the comma in the domain of a polarized polycategory in case $\mathbf{X}$ we have polarized polynatural (in everything outside A and B) equivalences:

$$\begin{array}{c} \Gamma, A, B, \Gamma' \mathrel{\underset{o\ p}{\mid\!\!\longrightarrow}} \Delta \\ \hline\hline \Gamma, A \otimes B, \Gamma' \mathrel{\underset{o\ p}{\mid\!\!\longrightarrow}} \Delta \end{array} \qquad \begin{array}{c} \Gamma, A, B, \Gamma' \vdash_o \Delta \,/\, X \,\backslash\, \Delta' \\ \hline\hline \Gamma, A \otimes B, \Gamma' \vdash_o \Delta \,/\, Y \,\backslash\, \Delta' \end{array}$$

$$\begin{array}{c} \Gamma, A, B, \Gamma' \,/\, Y \,\backslash\, \Gamma'' \vdash_p \Delta \\ \hline\hline \Gamma, A \otimes B, \Gamma' \,/\, Y \,\backslash\, \Gamma'' \vdash_p \Delta \end{array} \quad \begin{array}{c} \Gamma \,/\, Y \,\backslash\, \Gamma', A, B, \Gamma'' \vdash_p \Delta \\ \hline\hline \Gamma \,/\, Y \,\backslash\, \Gamma', A \otimes B, \Gamma'' \vdash_p \Delta \end{array}$$

Similarly we may represent an empty domain by an object $\top$ which induces polynatural equivalences (in everything):

$$\begin{array}{c} \Gamma, \Gamma' \mathrel{\underset{o\ p}{\mid\!\!\longrightarrow}} \Delta \\ \hline\hline \Gamma, \top, \Gamma' \mathrel{\underset{o\ p}{\mid\!\!\longrightarrow}} \Delta \end{array} \qquad \begin{array}{c} \Gamma, \Gamma' \vdash_o \Delta \,/\, X \,\backslash\, \Delta' \\ \hline\hline \Gamma, \top, \Gamma' \vdash_o \Delta \,/\, Y \,\backslash\, \Delta' \end{array}$$

$$\begin{array}{c} \Gamma, \Gamma' \,/\, Y \,\backslash\, \Gamma'' \vdash_p \Delta \\ \hline\hline \Gamma, \top, \Gamma' \,/\, Y \,\backslash\, \Gamma'' \vdash_p \Delta \end{array} \quad \begin{array}{c} \Gamma \,/\, Y \,\backslash\, \Gamma', \Gamma'' \vdash_p \Delta \\ \hline\hline \Gamma \,/\, Y \,\backslash\, \Gamma', \top, \Gamma'' \vdash_p \Delta \end{array}$$

Dually we shall say that an object $Y \oplus Z$ (co)represents the comma in the codomain in case we have polarized polynatural (in everything outside Y and Z) equivalences:

$$\begin{array}{c} \Gamma \mathrel{\underset{o\ p}{\mid\!\!\longrightarrow}} \Delta, Y, Z, \Delta' \\ \hline\hline \Gamma \mathrel{\underset{o\ p}{\mid\!\!\longrightarrow}} \Delta, Y \oplus Z, \Delta' \end{array} \qquad \begin{array}{c} \Gamma \,/\, A \,\backslash, \Gamma' \vdash_o \Delta, Y, Z, \Delta' \\ \hline\hline \Gamma \,/\, A \,\backslash\, \Gamma' \vdash_o \Delta, Y \oplus Z, \Delta' \end{array}$$

$$\begin{array}{c} \Gamma \vdash_p \Delta, X, Y, \Delta' \,/\, A \,\backslash\, \Delta'' \\ \hline\hline \Gamma \vdash_p \Delta, X \oplus Y, \Delta' \,/\, A \,\backslash\, \Delta'' \end{array} \quad \begin{array}{c} \Gamma \vdash_p \Delta \,/\, A \,\backslash\, \Delta', Y, Z, \Delta'' \\ \hline\hline \Gamma \vdash_p \Delta \,/\, A \,\backslash\, \Delta', Y \oplus Z, \Delta'' \end{array}$$

We shall say that a polarized polycategory is $\otimes$-representable in case for each pair of opponent objects A and B there is a representing object $A \otimes B$ as above, and the empty domain also has a representing object $\top$. Dually if for each pair of player objects Y and Z there is a (co)representing player object $Y \oplus Z$ and the empty codomain is (co)represented (in the obvious dual manner) by and object $\perp$ then the polarized polycategory is $\oplus$-representable.

Similarly we may define the representability for the context binders to be provided by objects with polarized polynatural equivalences:

$$\begin{array}{c} \Gamma, X \,/\, A \,\backslash\, \Gamma' \vdash_p \Delta \\ \hline\hline \Gamma \,/\, X \otimes\!\rangle\, A \,\backslash\, \Gamma' \vdash_p \Delta \end{array} \quad \begin{array}{c} \Gamma \,/\, A \,\backslash\, X, \Gamma' \vdash_p \Delta \\ \hline\hline \Gamma \,/\, A \,\langle\!\otimes X \,\backslash\, \Gamma' \vdash_p \Delta \end{array}$$

$$\begin{array}{c} \Gamma \vdash_o \Delta, A \,/\, X \,\backslash\, \Delta'' \\ \hline\hline \Gamma \vdash_o \Delta \,/\, A \oplus\!\rangle\, X \,\backslash\, \Delta'' \end{array} \quad \begin{array}{c} \Gamma \vdash_o \Delta \,/\, X \,\backslash\, A, \Delta'' \\ \hline\hline \Gamma \vdash_o \Delta \,/\, X \,\langle\!\oplus A \,\backslash\, \Delta'' \end{array}$$

We shall say that a polarized polycategory is representable in case there are representing objects for *all* the separators in the logic.

The ability to represent the separators in the sequent calculus allows one to compress the content of the polarized polycategorical structure into the polarized categorical structure determined by the mere arrows (*i.e.* polyarrows with just one source and one target). This is because any polyarrow, once all its separators have been represented, corresponds (possibly in many ways) to a representing arrow. The polarized polycategorical structure forces functorial structure which must satisfy various coherence conditions onto the polarized category determined by its mere arrows. This structure will be explored in section 6.6. In particular, it is possible to provide a complete axiomatization at the polarized categorical level of the structure which can occur as the arrows of a representable polarized polycategories.

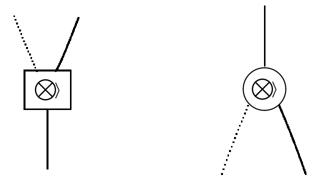
Various aspects of this structure is readily apparent: for example the tensor and par are both part of monoidal structures on their respective categories: the associativity follows from the implicit associativity of the polycategorical separators. A little less obvious are the linear distributivities (which are not isomorphisms) but mediate between the behavior of the separators in the domain and those in the codomain of a polarized polycategory. Here is an example of a derivation of a linear distribution for this system:

$$\dfrac{\dfrac{\dfrac{\dfrac{1_{B\oplus C}}{/\, B \oplus C \,\backslash \vdash_p B \oplus C}}{/\, B\oplus C \,\backslash \vdash_p B, C} \qquad \dfrac{\dfrac{1_{A\otimes B}}{/\, A\otimes B\,\backslash \vdash_p A\otimes B}}{A \,/\, B \,\backslash \vdash_p A \otimes B}}{A\,/\,B\oplus C\,\backslash \vdash_p A\otimes B, C}\,\text{Cut}}{/\, A\otimes(B\oplus C)\,\backslash \vdash_p (A\otimes B)\oplus C}$$

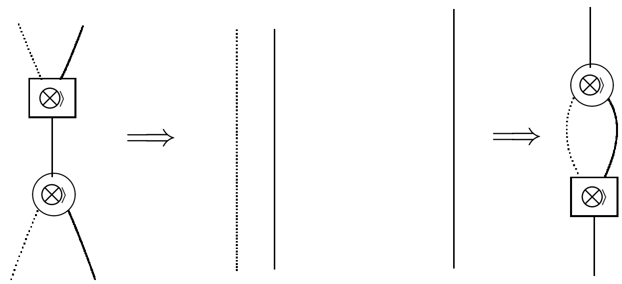
6.5. Proof circuits for multiplicatives. Representablity can be directly expressed in the circuits for polarized polyarrows. Each of the multiplicatives, by reversing the representation of the identity arrow on the multiplicative type, gives rise to a component. We shall indicate components by enclosing them in square boxes.

Somewhat more difficult is the effect of representing in the direction of using the equivalence to introduce a multiplicative type. In order introduce a type in this manner one already needs a legal circuit onto which one adds a link (these are referred to as switching links and unit thinning links) to the type which is to be represented. One may view these links in various ways. Formally they are actually scoping devices, however, the polynaturality of representation dictates that the scope can be expanded and contracted quite freely. It is therefore more usual not to represent the scope directly in the circuit but rather to require quite separately that the circuit has a valid scoping: which is achieved by checking a "net condition".

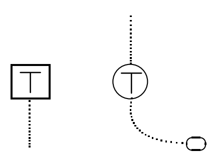
For the representation of each binary multiplicative we have therefore a component and a link. Each has a corresponding reduction rule and an expansion rule. For example, the component and link for the introduction and elimination rules for the mixed tensor $\otimes$ are the following; the other connectives are similar, with appropriate typing of the wires.



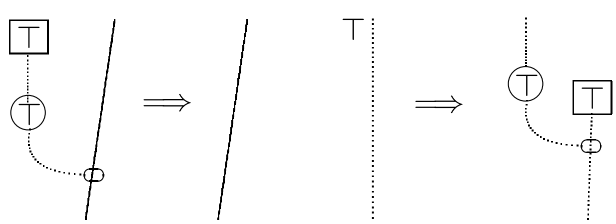
The corresponding reduction and elimination rewrites are these:



For each unit there is a component and a link. The scope of a unit link is indicated by connecting it by a (movable) thinning link to another part of the circuit. For example, here are the component and link for the tensor unit $\top$. (The small oval at the end of the thinning link which exits from the $\top$ link is used to indicate where the thinning link is attached to some other wire in the circuit.)



The reduction and expansion rewrites for $\top$ are these:



It is the behavior of the units which provides almost all the complexity to the situation and as we shall shortly see this complexity is directly inherited from the unpolarized version.

We represent links as round nodes. The net condition can be most simply expressed by requiring that the circuit has an inductive construction from components using cut and scoped introduction of links.

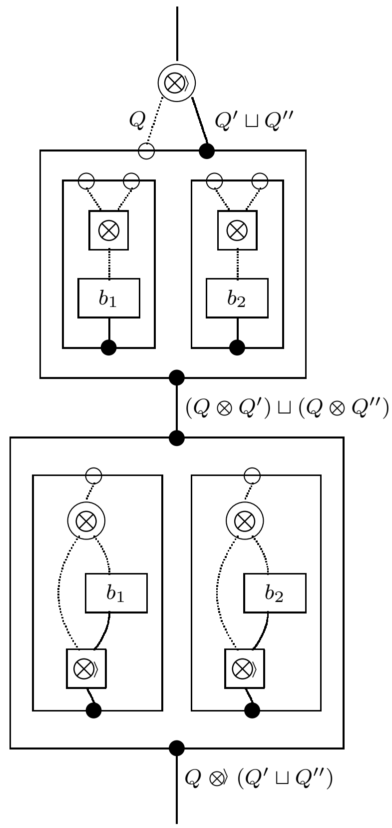
As an example of the circuits including the tensor structure, we illustrate the isomorphism

$$Q \otimes (Q' \sqcup Q'') \cong (Q \otimes Q') \sqcup (Q \otimes Q'')$$

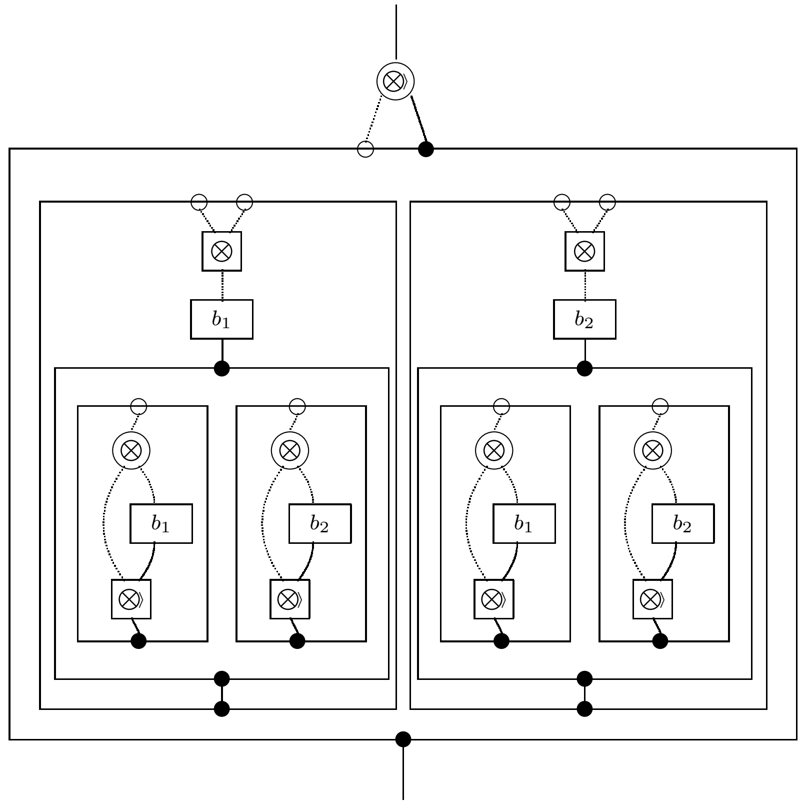
as given by the derivations (8) in section 6.3. We start with the composite

$$Q \otimes\!\rangle\, (Q' \sqcup Q'') \longrightarrow (Q \otimes Q') \sqcup (Q \otimes Q'') \longrightarrow Q \otimes\!\rangle\, (Q' \sqcup Q'')$$

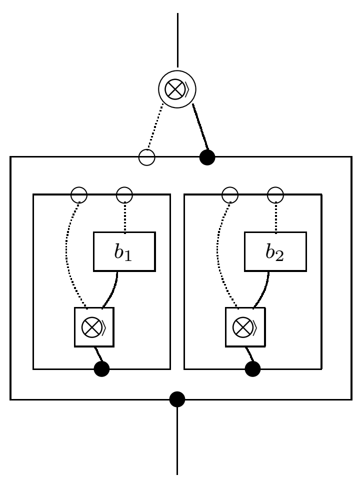
given by the following circuit (in its unreduced form). (We omit most of the labels on wires, since they may easily be reconstructed by the reader.)



This reduces to the following circuit.

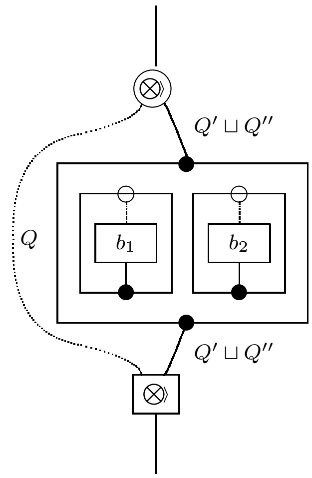


which in turn reduces to



This circuit is in expanded normal form — it is the expanded normal form for the identity on $Q \otimes (Q' \sqcup Q'')$. There is a curiosity here; using the usual expansions on a $\sqcup$ wire and a

$\otimes\!\!)$ wire, we see that the identity on $Q \otimes\!\!) \, (Q' \sqcup Q'')$ expands to this.

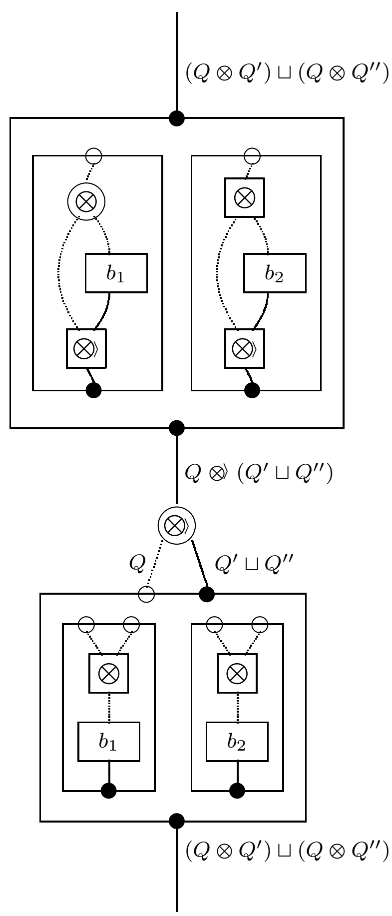


but this clearly admits a reduction (to the expanded normal form displayed above).

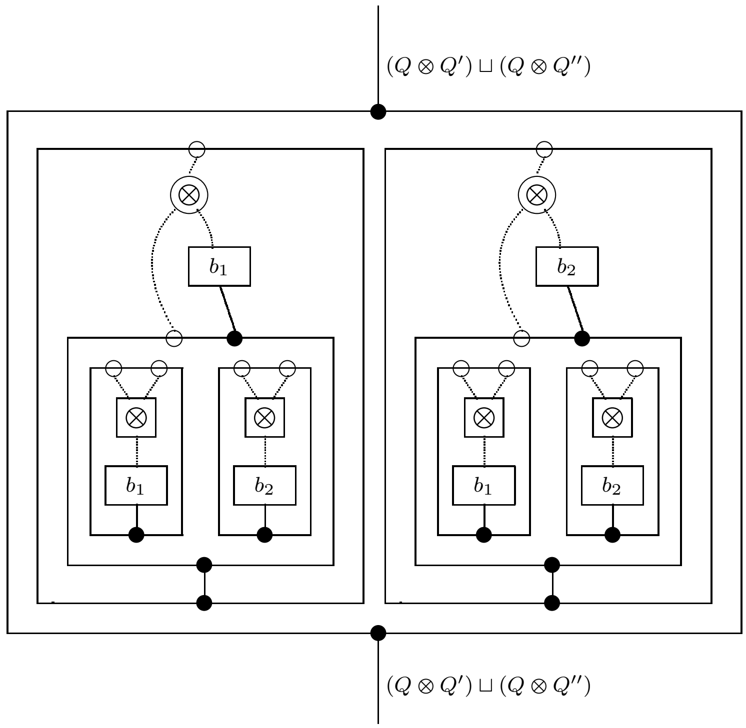
For the converse direction, we must show that

$$(Q \otimes Q') \sqcup (Q \otimes Q'') \longrightarrow Q \otimes\!\!) \, (Q' \sqcup Q'') \longrightarrow (Q \otimes Q') \sqcup (Q \otimes Q'')$$

is the identity on $(Q \otimes Q') \sqcup (Q \otimes Q'')$. The composite is the following circuit.



which reduces to the following circuit, which clearly then reduces to the expanded normal form of the identity on $(Q \otimes Q') \sqcup (Q \otimes Q'')$.



Recall that in any (closed) free polarized polycategory, every component of a focused polyarrow must be focused, whereas in an unfocused polyarrow exactly one component is unfocused (see Lemmas 4.3.2 and 6.2.3). In order to state the analogous result for free representable polarized polycategories we must be clear about what polyfunctors are involved. In fact, we shall use the same notion of polyfunctor as in section 5.1, so we shall simply ignore the links. In other words, representability is regarded as purely coincidental structure. These are the morphism of [CKS].

Given the circuit calculus above (leaving the additives aside for the moment) it is easy to see that:

6.5.1. Proposition. *The underlying functor* $U\colon \mathbf{PolPolyCat}_{\mathsf{rep}} \longrightarrow \mathbf{PolPolyGraph}$*, from representable polarized polycategories to polarized polygraphs, has a left adjoint*

$$F_{\mathsf{rep}}\colon \mathbf{PolPolyGraph} \longrightarrow \mathbf{PolPolyCat}_{\mathsf{rep}}$$

We now have:

6.5.2. Lemma.

1. *In any focused circuit of* $F_{\mathsf{rep}}(\mathbf{G})$ *every component is focused,*
2. *In any unfocused circuit of* $F_{\mathsf{rep}}(\mathbf{G})$ *there is at most one unfocused component.*

Proof. All the multiplicative components are focused, thus any polyarrow which does not include any links satisfies both conditions. It remains to check the effect of adding a link: the links are also focused and, therefore preserve the focused or unfocused nature of the polyarrows they make representable. This allows the inductive step for links to be made for each part of this lemma. ∎

As before we may ask whether, given an unpolarized circuit for a polyarrow in the free representable polycategory, it is always possible to provide a polarization of the arrow (if polarized versions of all the components are provided). Unlike before we notice that, while it is certainly always possible to provide a polarization for any (external) focused polarization, the steps where links are added to the circuit allow choice, so the polarization will not be uniquely determined by the external focused polarization.

As before we shall use a structural induction on the circuit. Assume inductively for a given circuit that for any specified external polarization it is possible to find a polarization of the components and links which satisfies the specification. The interesting case to be considered is whether when we add a binary link to the circuit it still satisfies the desired property. There are two cases to consider: when the binary link will provide an interface wire which is required to be in focus and when this is not the case. In the latter case the typing of the link is determined and this provides a unique polarization specification for the subcircuit. In the former case, however, we have two possible context binding links we can apply: this means that there are two possible polarization specifications which we could require the subcircuit to satisfy. This means that there are now at least two solutions.

6.5.3. Corollary. *In the free representable polycategory on components:*

1. *Each proof circuit presentation of a polyarrow can be polarized to satisfy any specified external focused polarization in possibly many different ways (bounded by 2^n where n is the number of switching links).*

2. *Each proof circuit presentation of a polyarrow, which has at least one component whose polarization can be unfocused, can be polarized to be an unfocused polyarrow in possibly many different ways.*

An important consequence of this is that deciding whether two maps are equal (or are present) for the polarized multiplicatives is just as hard as in the unpolarized case. Thus, polarizing does not make the decision problems for the pure multiplicative structure any less intractable.

6.6. Linear polarized categories. The mere arrows of a representable polarized polycategory form a **linear polarized category**. As mentioned earlier, this structure can be axiomatized in terms of its functorial structure, although this axiomatization involves a large number of diagrams. Rather than giving a complete documentation of these diagrams we shall describe how they can be generated and why they provide a complete axiomatization of the structure determined by the mere arrows of representable polycategories.

First we provide a description of the functorial structure: we already know that it is necessary to introduce six functorial operations (corresponding to the two tensors and their four context binders) and two units. These can be introduced as polarized functors:

$$
\begin{tikzcd}
\mathbf{X}_o \times \mathbf{X}_o \arrow[r, "\widehat{X}\times 1", "|"{marking}] \arrow[d, "\otimes"'] \arrow[dr, phantom, "\Downarrow \widehat{\langle\!\otimes}"] & \mathbf{X}_p \times \mathbf{X}_o \arrow[d, "\langle\!\otimes"] \\
\mathbf{X}_o \arrow[r, "\widehat{X}"', "|"{marking}] & \mathbf{X}_p
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathbf{X}_o \times \mathbf{X}_o \arrow[r, "1\times\widehat{X}", "|"{marking}] \arrow[d, "\otimes"'] \arrow[dr, phantom, "\Downarrow \widehat{\otimes\!\rangle}"] & \mathbf{X}_o \times \mathbf{X}_p \arrow[d, "\otimes\!\rangle"] \\
\mathbf{X}_o \arrow[r, "\widehat{X}"', "|"{marking}] & \mathbf{X}_p
\end{tikzcd}
$$

$$
\begin{tikzcd}
\mathbf{X}_o \times \mathbf{X}_p \arrow[r, "\widehat{X}\times 1", "|"{marking}] \arrow[d, "\langle\!\oplus"'] \arrow[dr, phantom, "\Downarrow \widehat{\langle\!\oplus}"] & \mathbf{X}_p \times \mathbf{X}_p \arrow[d, "\oplus"] \\
\mathbf{X}_o \arrow[r, "\widehat{X}"', "|"{marking}] & \mathbf{X}_p
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathbf{X}_p \times \mathbf{X}_o \arrow[r, "1\times\widehat{X}", "|"{marking}] \arrow[d, "\oplus\!\rangle"'] \arrow[dr, phantom, "\Downarrow \widehat{\oplus\!\rangle}"] & \mathbf{X}_p \times \mathbf{X}_p \arrow[d, "\oplus"] \\
\mathbf{X}_o \arrow[r, "\widehat{X}"', "|"{marking}] & \mathbf{X}_p
\end{tikzcd}
$$

These are subject to several coherence conditions, described below. But before we describe the coherence conditions in more detail, let us consider the meaning of the diagrams above (they look like natural transformations, but really they are polarized functors). Consider for example the second one, involving $\otimes\!\rangle$. Any (ordinary) category induces a polarized category with the identity module (so all arrows are module arrows). Both $\mathbf{X}_o$ and $\mathbf{X}_p$ may be considered polarized in this way, and so we have polarized categories $\mathbf{X} \times \mathbf{X}_o$ and $\mathbf{X}_o \times \mathbf{X}$ (as well as $\mathbf{X} \times \mathbf{X}_p$ and $\mathbf{X}_p \times \mathbf{X}$). $\langle \otimes, \otimes\!\rangle, \widehat{\otimes\!\rangle} \rangle$ is a polarized tensor, *viz.* a polarized functor $\mathbf{X}_o \times \mathbf{X} \longrightarrow \mathbf{X}$, where $\otimes$ is an ordinary tensor on $\mathbf{X}_o$, $\otimes\!\rangle$ is a bifunctor $\mathbf{X}_o \times \mathbf{X}_p \longrightarrow \mathbf{X}_p$, and $\widehat{\otimes\!\rangle}$ is a function which, given arrows $f\colon A_1 \longrightarrow A_1'$ in $\mathbf{X}_o$ and $m\colon A_2 \relbar\joinrel\mapstochar\joinrel\rightarrow B_2$ in $\widehat{\mathbf{X}}$ assigns an arrow $f \widehat{\otimes\!\rangle} m\colon A_1 \otimes A_2 \relbar\joinrel\mapstochar\joinrel\rightarrow A_1' \otimes\!\rangle B_2$ in $\widehat{\mathbf{X}}$. Similarly, the fourth diagram represents a polarized tensor which, on the module, acts as follows: given arrows $g\colon B_1 \longrightarrow B_1'$ in $\mathbf{X}_p$ and $m\colon A_2 \relbar\joinrel\mapstochar\joinrel\rightarrow B_2$ in $\widehat{\mathbf{X}}$, we have an arrow $g \widehat{\oplus\!\rangle} m\colon B_1 \oplus\!\rangle A_2 \relbar\joinrel\mapstochar\joinrel\rightarrow B_1' \oplus B_2$ in $\widehat{\mathbf{X}}$. The other two diagrams are dual. Note that in terms of the game logic with channels, these are special cases of the appropriate introduction rules for these type constructors (and indeed, these simple cases are sufficient to generate all others using the cut rules). Of course, these are "decorated" versions of simple functoriality of the non-polarized tensor and par.

The coherence diagrams these functors must satisfy correspond precisely to all the polarized versions of the conditions for linearly distributive categories [CS97a]. To illustrate this let us consider some concrete examples. Consider for example the following generalizations of the diagrams given in [CS97a, pp. 147–148].

$$
\begin{tikzcd}
\top \otimes\!\rangle (B \oplus B') \arrow[d, "\delta"'] \arrow[dr, "u"] & \\
(\top \otimes\!\rangle B) \oplus B' \arrow[r, "u \oplus 1"'] & B \oplus B'
\end{tikzcd}
\qquad
\begin{tikzcd}[column sep=tiny]
 & A_1 \otimes (A_2 \otimes (A_3 \langle\!\oplus B)) \arrow[dl, "a^{-1}"'] \arrow[dr, "1 \otimes \delta"] & \\
(A_1 \otimes A_2) \otimes (A_3 \langle\!\oplus B) \arrow[d, "\delta"'] & & A_1 \otimes ((A_2 \otimes A_3) \langle\!\oplus B) \arrow[d, "\delta"] \\
((A_1 \otimes A_2) \otimes A_3) \langle\!\oplus B \arrow[rr, "a \langle\!\oplus 1"'] & & (A_1 \otimes (A_2 \otimes A_3)) \langle\!\oplus B
\end{tikzcd}
$$

where we denote by δ the natural transformations

$$A \otimes (B_1 \oplus B_2) \longrightarrow (A \otimes B_1) \oplus B_2$$

$$\text{and its dual} \quad A_1 \otimes (A_2 \oplus B) \longrightarrow (A_1 \otimes A_2) \oplus B$$

that are two of the "decorated" versions of the usual linear distributivity, u for the natural isomorphism $\top \otimes B \longrightarrow B$ corresponding to the tensor unit, and a for the associativity of tensor. These and similar conditions are required, as well as ("decorated" versions of) naturality conditions, such as the following, for arrows $f\colon A_1 \longrightarrow A_1'$ in $\mathbf{X}_o$, $g\colon A_2 \longrightarrow\!\!\!\!\!\!\!+\;\; B_2$ in $\widehat{\mathbf{X}}$, and $h\colon B \longrightarrow B_3$ in $\mathbf{X}_p$.

$$\begin{tikzcd}
A_1 \otimes (A_2 \oplus B) \arrow[r, "f\otimes(g\oplus h)", "+"{marking}] \arrow[d, "\delta"'] & A_1' \otimes (B_2 \oplus B_3) \arrow[d, "\delta"] \\
(A_1 \otimes A_2) \oplus B \arrow[r, "(f\otimes g)\oplus h"', "+"{marking}] & (A_1' \otimes B_2) \oplus B_3
\end{tikzcd}$$

Basically, all diagrams (*i.e.* decorated well-typed versions of the standard commutative diagrams for linearly distributive categories) commute.

In order to show that these diagrams axiomatize linear polarized categories we must do two things: show that the every linear polarized category satisfies these axioms and show that any polarized category equipped with this structure can be viewed as a representable polarized polycategory. The first step is straight forward, if lengthy; one can easily check that the axioms are true in any free representable category.

For the converse we build a representable polarized polycategory, $\mathbf{Poly}(\mathbf{X})$ from a polarized category $\mathbf{X}$ equipped with this functorial structure. To obtain a polarized polycategory we must describe the homsets $\mathbf{Poly}(\mathbf{X})_o(\Gamma\ ;\ \Delta \,/\, X \setminus \Delta')$, $\mathbf{Poly}(\mathbf{X})_p(\Gamma \,/\, Y \setminus \Gamma'\ ;\ \Delta)$, and $\mathbf{Poly}(\mathbf{X})_{op}(\Gamma\ ;\ \Delta)$ where Γ, Γ', Δ, and Δ' are strings. This we do by choosing a particular representation of the lists:

$$\begin{aligned}
[\![\Gamma]\!] &= [\![X_1, \ldots, X_n]\!] = X_1 \otimes (X_2 \otimes (\ldots (X_n \otimes \top) \ldots)) \\
[\![\Delta]\!] &= [\![Y_1, \ldots, Y_m]\!] = Y_1 \oplus (Y_2 \oplus (\ldots (Y_m \oplus \bot) \ldots)) \\
[\![\Gamma \,/\, X \setminus \Gamma']\!] &= ([\![\Gamma]\!] \otimes X) \otimes [\![\Gamma']\!] \\
[\![\Delta \,/\, X \setminus \Delta']\!] &= ([\![\Delta]\!] \oplus X) \oplus [\![\Delta']\!]
\end{aligned}$$

and setting

$$\begin{aligned}
\mathbf{Poly}(\mathbf{X})_o(\Gamma\ ;\ \Delta \,/\, X \setminus \Delta') &\equiv \mathbf{X}_o([\![\Gamma]\!], [\![\Delta \,/\, X \setminus \Delta']\!]) \\
\mathbf{Poly}(\mathbf{X})_p(\Gamma \,/\, Y \setminus \Gamma'\ ;\ \Delta) &\equiv \mathbf{X}_p([\![\Gamma \,/\, Y \setminus \Gamma']\!], [\![\Delta]\!]) \\
\mathbf{Poly}(\mathbf{X})_{op}(\Gamma\ ;\ \Delta) &\equiv \mathbf{X}_{op}([\![\Gamma]\!]\ ;\ [\![\Delta]\!])
\end{aligned}$$

Almost by definition this will give a setting which is representable. The remaining difficulty is to show how the polycategorical composition given by cut is defined. This is where the (24 different!) polarizations of the linear distributions play a crucial role: they provide the prototypical cuts from which any cuts can be obtained.

The proof we have sketched (which follows the proof for linearly distributive categories) allows us to conclude:

6.6.1. Theorem. *A polarized category equipped with the functorial structure satisfying all the polarized axioms of a linearly distributive category is a linear polarized category.*

It remains to specify what happens regarding the additives. A **linear polarized game category** is a linear polarized category with polarized products together with the requirement that "tensor preserves coproducts" and "par preserves products" in the sense that:

$$\coprod_{i\in I} Q \otimes Q_i \cong Q \otimes\!\rangle \left(\coprod_{i\in I} Q_i\right) \qquad \coprod_{i\in I} Q_i \otimes Q \cong \left(\coprod_{i\in I} Q_i\right) \langle\!\otimes\, Q$$

where the isomorphism is given by cotupling the tensored coprojections and

$$P \oplus\!\rangle \left(\prod_{j\in J} P_i\right) \cong \prod_{j\in J} P \oplus P_j \qquad \left(\prod_{j\in J} P_j\right) \langle\!\oplus\, P \cong \prod_{j\in J} P_j \oplus P$$

where the isomorphisms are by tupling the pared projections.

In the unary case this give the effect on the inner structure:

$$(Q \otimes X)^* \cong Q \otimes\!\rangle\, X^* \qquad (X \otimes Q)^* \cong X^* \langle\!\otimes\, Q$$

$$P \oplus\!\rangle\, Y_* \cong (P \oplus Y)_* \qquad Y_* \langle\!\oplus\, Q \cong (Y \otimes Q)_*$$

For aft additives the situation is similar but there is no polarity change so the isomorphisms look like:

$$\bigvee_{i\in I} Q \otimes\!\rangle\, P_i \cong Q \otimes\!\rangle \left(\bigvee_{i\in I} P_i\right) \qquad \bigvee_{i\in I} P_i \langle\!\otimes\, Q \cong \left(\bigvee_{i\in I} P_i\right) \langle\!\otimes\, Q$$

$$\bigwedge_{j\in J} P \oplus\!\rangle\, Q_j \cong P \oplus\!\rangle \left(\bigwedge_{j\in J} Q_j\right) \qquad \bigwedge_{j\in J} Q_j \langle\!\oplus\, P \cong \left(\bigwedge_{j\in J} Q_j\right) \langle\!\oplus\, P$$

6.6.2. Example. Of course, our main example is AJ games, and we shall see in section 7 that they do form a representable polarized game polycategory, but for now, we shall briefly indicate another example we have been considering may be representable. Example 3.2.4, $\mathcal{G}(\mathbf{C},\mathcal{K})$, is representable if $\mathbf{C}$ is representable (and so is a linearly distributive category), and if we impose some special properties on I, J. Specifically, suppose that I carries a $\otimes$-comonoid structure $\top \xleftarrow{\epsilon} \mathsf{I} \xrightarrow{\delta} \mathsf{I} \otimes \mathsf{I}$, which dualizes (under negation) to a $\oplus$-monoid structure on J: $\bot \xrightarrow{\eta} \mathsf{J} \xleftarrow{\mu} \mathsf{J} \oplus \mathsf{J}$. Then we can take (for $\otimes$ in $\mathcal{G}(\mathbf{C},\mathcal{K})$)

$$(\mathcal{R}, X) \otimes (\mathcal{R}', X') = (\{\delta\,;\, r \otimes r' \mid r \in \mathcal{R}, r' \in \mathcal{R}'\}, X \otimes X')$$

with tensor unit $(\{\epsilon\}, \top)$. For the "cross tensor" we take

$$(\mathcal{R}, X) \otimes\!\rangle\, (Y, \mathcal{S}) = (X \otimes Y, \{k\colon X \times Y \longrightarrow \mathsf{J} \mid \delta\,;\, r \otimes r' \perp k \ \forall r' \in \mathcal{S}_*\})$$

This definition is in fact forced by the representability condition, so it is now straightforward to show that $\mathcal{G}(\mathbf{C},\mathcal{K})$ is representable.

6.7. Adding aft additives and representability. We have described linear polarized categories with additives and the categorical properties we expect to hold. But what effect does the game construction have on representability? We might be extremely lucky: it might be the case that applying the game construction to a representable polarized polycategory will always result in a representable polarized game polycategory. But unfortunately this is not the case, although we do get surprisingly close!

It is useful to start with the addition of aft additives in order to see what can be expected. Recall that in Proposition 5.4.1 we showed how to construct the free polarized polycategory with additives, $F_{\vee\wedge}(\mathbf{X})$ from an arbitrary polarized polycategory $\mathbf{X}$. We observe:

6.7.1. Proposition. *If $\mathbf{X}$ is a $\otimes\oplus$-representable polarized polycategory then so is $F_{\vee\wedge}(\mathbf{X})$.*

Proof. Let us use the notation $[\![\Gamma]\!]$ to denote the object representing Γ. Then we define the representing object (*i.e.* the tensor) for a pair of opponent objects to be:

$$\left[\!\!\left[\left(\bigwedge_{i\in I} X_i\right) \otimes \left(\bigwedge_{j\in J} X'_j\right)\right]\!\!\right] = \bigwedge_{(i,j)\in I\times J} [\![X_i]\!] \otimes [\![X'_j]\!]$$

We then have to provide the polynatural equivalences between the polyhom-sets. Consider

$$\frac{\Gamma, \bigwedge_{i\in I} X_i, \bigwedge_{j\in J} X'_j, \Gamma' \underset{o\ p}{\mapsto} \Delta \text{ in} F_{\vee\wedge}(\mathbf{X})_{op}}{\overline{\Gamma, \bigwedge_{(i,j)\in I\times J} X_i \otimes X'_j, \Gamma' \underset{o\ p}{\mapsto} \Delta \text{ in} F_{\vee\wedge}(\mathbf{X})_{op}}}$$

An element in the homset is given by a pair (x, h) where $x = (i_1, \ldots, i, j, \ldots ; \ldots, j_m)$ and h is polyarrow in $\mathbf{X}$ of type $\mathbf{X}(X_{i_1,1}, \ldots, X_i, X'_j, \ldots ; \ldots Y_{j_{m,m}})$. Using representability in $\mathbf{X}$ we have:

$$\frac{h \in \mathbf{X}(X_{i_1,1}, \ldots, X_i, X'_j, \ldots ; \ldots Y_{j_{m,m}})}{\overline{h' \in \mathbf{X}(X_{i_1,1}, \ldots, X_i \otimes X'_j, \ldots ; \ldots Y_{j_{m,m}})}}$$

so that we have a bijective correspondence to elements of the form (x', h') where h' is as above and $x' = (i_1, \ldots, (i, j), \ldots ; \ldots, j_m)$. It is clear that the polynaturality of this follows from the polynaturality of the representation in $\mathbf{X}$.

The argument for focused polyhom-sets follows exactly the same lines: this time the map is given by a finite span in Sets from the focus and a family of polyarrows. The span is replaced by a span in which the two legs being represented are paired to produce a map to the product and the assignment is modified using representability in $\mathbf{X}$ as above. ∎

Notice that the final polarized polycategory 1 is representable and, therefore, $F_{\vee\wedge}(1)$ is $\otimes\oplus$-representable. However, notice that $F_{\vee\wedge}(1)$ is *not* representable with respect to the context binders ($\otimes\!\!\!\rangle$, $\langle\!\!\!\oplus$, *etc.*). This is an indication that we cannot expect that in general every aspect of representability will lift through these constructions.

6.7.2. Proposition. *Given a polarized polycategory $\mathbf{X}$:*

1. *if $\otimes, \langle\!\!\!\otimes, \otimes\!\!\!\rangle$ (respectively $\oplus, \langle\!\!\!\oplus, \oplus\!\!\!\rangle$) are representable in $\mathbf{X}$ then $\otimes$ (respectively $\oplus$) is representable in $\mathsf{Gam}(\mathbf{X})$;*
2. *if $\mathbf{X}$ has $\otimes, \langle\!\!\!\otimes, \otimes\!\!\!\rangle$ symmetric (respectively $\oplus, \langle\!\!\!\oplus, \oplus\!\!\!\rangle$ symmetric) then $\otimes$ is symmetric (respectively $\oplus$ symmetric) in $\mathsf{Gam}(\mathbf{X})$;*

Proof. The proof of this proposition is essentially an extension of the Whitman theorems of section 3. We shall indicate the highlights.

1. To represent $\otimes$ in $\mathsf{Gam}(\mathbf{X})$ we may inductively define

$$\left[\!\!\left[\prod_{i\in I} P_i \otimes \prod_{j\in J} P'_j\right]\!\!\right] = \prod_{\substack{i\in I\\ P_i \text{ not atomic}}} \left[\!\!\left[P_i \mathbin{(\!\otimes} \prod_j P'_j\right]\!\!\right] \wedge \prod_{\substack{j\in J\\ P'_j \text{ not atomic}}} \left[\!\!\left[\prod_i P_i \mathbin{\otimes\!)} P'_j\right]\!\!\right]$$

$$\left[\!\!\left[\coprod_{i\in I} Q_i \mathbin{(\!\otimes} Q\right]\!\!\right] = \coprod_{i\in I} [\![Q_i \otimes Q]\!]$$

$$\left[\!\!\left[\prod_{i\in I} P_i \otimes Q\right]\!\!\right] = \prod_{i\in I} [\![P_i \mathbin{(\!\otimes} Q]\!]$$

and dually for $\otimes\!)$. The only subtle point is that if P is atomic then there are no "new" sequents in $\mathsf{Gam}(\mathbf{X})$ of the form $\Gamma \,/\, P \setminus \Gamma' \vdash_o \Delta$ other than those in $\mathbf{X}$. So we do not want to represent $P \mathbin{(\!\otimes} \prod_i P_i$ or $\prod_i P_i \mathbin{\otimes\!)} P$ for atomic P. There is no problem when both objects are atomic, as then we are in $\mathbf{X}$. $\oplus$ is treated similarly.

2. The only point to be made here is just what is meant by "$(\!\otimes, \otimes\!)$ are symmetric": clearly this just means that there is a coherent natural isomorphism $P \mathbin{(\!\otimes} Q \cong Q \mathbin{\otimes\!)} P$. That such symmetry (including of $\otimes$) lifts to $\mathsf{Gam}(\mathbf{X})$ is obvious. ∎

6.8. Adding fore additives and representability. Next we consider the effect of freely adding fore additives; it turns out that we can lift representability for all the tensors, including the mixed ones, provided we have aft additive structure to begin with.

6.8.1. Lemma.

1. *Aft additive structure in* $\mathbf{X}$ *lifts to* $\mathrm{Fam}(\mathbf{X})$*: if* $\mathbf{X}$ *is a polarized game polycategory with aft additives, then so is* $\mathrm{Fam}(\mathbf{X})$.

2. *Tensor and par also lift to* $\mathrm{Fam}(\mathbf{X})$*: if* $\mathbf{X}$ *is a polarized game polycategory which is* $\otimes$ *(respectively* $\oplus$*) representable, then so is* $\mathrm{Fam}(\mathbf{X})$.

3. *If* $\mathbf{X}$ *is a polarized game polycategory with non polarized aft sums (in* $\mathbf{X}_p$*) which is* $\otimes, (\!\otimes, \otimes\!)$ *representable (respectively, with aft products (in* $\mathbf{X}_o$*) and which is* $\oplus, (\!\oplus, \oplus\!)$ *representable), then* $\mathrm{Fam}(\mathbf{X})$ *is* $\otimes, (\!\otimes, \otimes\!)$ *representable (respectively* $\oplus, (\!\oplus, \oplus\!)$ *representable).*

Proof. First, note that it is well-known (indeed, by construction) that applying the coproduct Fam construction to a category with products produces a category with both products and coproducts — in fact, an extensive category (see [C93]). To see this extends to the poly context we have, we can use the "expansion" trick from subsection 5.6 to handle the contexts. The following series of bijections sets out the necessary equivalence to prove that aft additive structure lifts to the family categories. Notational note: we use double horizontal lines to denote bijections of hom sets, similar to the use for bijections of sequent derivations, and $\sum, \prod$ for disjoint union and product in Sets; also $F = \{f\colon I \longrightarrow J \mid f(i) \in J_i\}$ where $J = \sum_{i \in I} J_i$.

$$
\begin{array}{c}
\mathbf{X}_p(\Gamma \,/ (\bigvee_{i\in I} X_{if(i)})_{f\in F} \setminus \Gamma' \vdash_p \Delta) \\
\hline\hline
\sum_{f\in F} \mathbf{X}_p(\Gamma \,/ \bigvee_{i\in I} X_{if(i)} \setminus \Gamma' \vdash_p \Delta) \\
\hline\hline
\sum_{f\in F} \prod_{i\in I} \mathbf{X}_p(\Gamma \,/\, X_{if(i)} \setminus \Gamma' \vdash_p \Delta) \\
\hline\hline
\prod_{i\in I} \sum_{j\in J_i} \mathbf{X}_p(\Gamma \,/\, X_{ij} \setminus \Gamma' \vdash_p \Delta) \\
\hline\hline
\prod_{i\in I} \mathbf{X}_p(\Gamma \,/ (X_{ij})_{j\in J_i} \setminus \Gamma' \vdash_p \Delta)
\end{array}
$$

which gives us the aft sum in $\mathrm{Fam}(\mathbf{X})$. Note that this is essentially the 'externalization' of the (hypothetical) isomorphism $\bigvee_{i\in I} \prod_{j\in J_i} X_{ij} \simeq \prod_{f\in F} \bigvee_{i\in I} X_{if(i)}$ ("hypothetical", since the left hand object doesn't actually exist as such in $\mathrm{Fam}(\mathbf{X})$).

Also, representing simple tensor and par in $\mathrm{Fam}(\mathbf{X})$ is straightforward: $(X_i)_I \otimes (X'_j)_J = (X_i \otimes X'_j)_{I\times J}$, and the dual for par. The mixed operators are somewhat trickier, however. For example, $(X_i)_I \mathbin{\otimes\!\rangle} (X'_j)_J = \bigvee_{i\in I}(X_i \mathbin{\otimes\!\rangle} X'_j)_{j\in J}$, as may be seen from these correspondences.

$$
\begin{array}{c}
\mathbf{X}_p(\Gamma, (X_i)_I \,/ (X'_j)_J \setminus \Gamma' \vdash_p \Delta) \\
\hline\hline
\prod_{i\in I} \mathbf{X}_p(\Gamma, X_i \,/ (X'_j)_J \setminus \Gamma' \vdash_p \Delta) \\
\hline\hline
\prod_{i\in I} \sum_{j\in J} \mathbf{X}_p(\Gamma, X_i \,/\, X'_{j_i} \setminus \Gamma' \vdash_p \Delta) \\
\hline\hline
\prod_{i\in I} \sum_{j\in J} \mathbf{X}_p(\Gamma \,/\, X_i \mathbin{\otimes\!\rangle} X'_{j_i} \setminus \Gamma' \vdash_p \Delta) \\
\hline\hline
\prod_{i\in I} \mathbf{X}_p(\Gamma \,/ (X_i \mathbin{\otimes\!\rangle} X'_j)_J \setminus \Gamma' \vdash_p \Delta) \\
\hline\hline
\mathbf{X}_p(\Gamma \,/ \bigvee_I (X_i \mathbin{\otimes\!\rangle} X'_j)_J \setminus \Gamma' \vdash_p \Delta)
\end{array}
$$

The case for the mixed pars is dual. ■

6.9. Remarks on representability for free inner polarized polycategories with additives. We now consider the problem of representing the multiplicatives of the free inner polarized polycategory with (fore, aft, and polarized) additives $\mathsf{Gam}_{\vee\wedge+\times}(\mathbf{X})$ on a representable polarized polycategory $\mathbf{X}$.

The main difficulty is to determine the representing objects for each construct. To this end we have provided Table 15 in which the description of representing objects for the tensor structure is given, the par structure being dual. The idea of the table is that from the syntactic form of the objects one can inductively build a new object purely from the structure of $\mathbf{X}$ and the inner and additive structure.

6.9.1. Theorem. *If $\mathbf{X}$ is a representable polarized polycategory then $\mathsf{Gam}_{\vee\wedge+\times}$ is representable.*

Proof. The essence of the inductive proof is contained in Table 15; we shall sketch the main details in a series of bijections of hom-sets. When there are more than one possible hom sets, we shall write $\mathbf{X}$, to mean whichever of $\mathbf{X}_o, \mathbf{X}_p, \widehat{\mathbf{X}}$ fit the typing.

Table 15: The representing objects for $\mathsf{Gam}_{\vee\wedge+\times}(\mathbf{X})$.

Note: In the following A and B are supposed to be atomic

$$\llbracket A \otimes B \rrbracket := A \otimes B$$

$$\begin{aligned}
\llbracket X \otimes \textstyle\sum_I X_i \rrbracket &:= \textstyle\sum_I \llbracket X \otimes X_i \rrbracket & \llbracket \textstyle\sum_I X_i \otimes X \rrbracket &:= \textstyle\sum_I \llbracket X_i \otimes X \rrbracket \\
\llbracket Y_* \otimes B \rrbracket &:= \llbracket Y \mathbin{\langle\!\otimes} B \rrbracket_* & \llbracket A \otimes Y_* \rrbracket &:= \llbracket A \mathbin{\otimes\!\rangle} Y \rrbracket_*
\end{aligned}$$

$$\llbracket X_* \otimes Y_* \rrbracket := \llbracket X \mathbin{\langle\!\otimes} Y_* \rrbracket \times \llbracket X_* \mathbin{\otimes\!\rangle} Y \rrbracket$$

$$\begin{aligned}
\llbracket \textstyle\bigwedge_I X_i \otimes Y \rrbracket &:= \textstyle\prod_I \llbracket X_i \otimes Y \rrbracket & \llbracket Y \otimes \textstyle\bigwedge_I X_i \rrbracket &:= \textstyle\prod_I \llbracket Y \otimes X_i \rrbracket \\
\llbracket A \mathbin{\otimes\!\rangle} B \rrbracket &:= A \mathbin{\otimes\!\rangle} B & \llbracket A \mathbin{\langle\!\otimes} B \rrbracket &:= A \mathbin{\langle\!\otimes} B \\
\llbracket X \mathbin{\otimes\!\rangle} \textstyle\bigvee_I X_i \rrbracket &:= \textstyle\bigvee_I \llbracket X \mathbin{\otimes\!\rangle} X_i \rrbracket & \llbracket \textstyle\bigvee_I X_i \mathbin{\langle\!\otimes} X \rrbracket &:= \textstyle\bigvee_I \llbracket X_i \mathbin{\langle\!\otimes} X \rrbracket \\
\llbracket Y' \mathbin{\otimes\!\rangle} Y^* \rrbracket &:= \llbracket Y' \otimes Y \rrbracket^* & \llbracket Y^* \mathbin{\langle\!\otimes} Y' \rrbracket &:= \llbracket Y \otimes Y' \rrbracket^* \\
\llbracket \textstyle\bigwedge_I X_i \mathbin{\otimes\!\rangle} Y \rrbracket &:= \textstyle\prod_I \llbracket X_i \mathbin{\otimes\!\rangle} Y \rrbracket & \llbracket Y \mathbin{\langle\!\otimes} \textstyle\bigwedge_I X_i \rrbracket &:= \textstyle\prod_I \llbracket Y \mathbin{\langle\!\otimes} X_i \rrbracket \\
\llbracket Y_* \mathbin{\otimes\!\rangle} B \rrbracket &:= \textstyle\prod_\emptyset & \llbracket A \mathbin{\langle\!\otimes} Y_* \rrbracket &:= \textstyle\prod_\emptyset \\
\llbracket \textstyle\sum_I X_i \mathbin{\otimes\!\rangle} \prod_J Y_j \rrbracket &:= \textstyle\bigvee_I \prod_J \llbracket X_i \mathbin{\otimes\!\rangle} Y_j \rrbracket & \llbracket \textstyle\prod_J Y_j \mathbin{\langle\!\otimes} \sum_I X_i \rrbracket &:= \textstyle\bigvee_I \prod_J \llbracket Y_j \mathbin{\langle\!\otimes} X_i \rrbracket
\end{aligned}$$

[$\sum_I X_i \otimes X = \sum_I (X_i \otimes X)$]

$$\begin{array}{c}
\mathbf{X}(\Gamma, \sum_I X_i \otimes X, \Gamma' \; ; \; \Delta) \\ \hline\hline
\mathbf{X}(\Gamma, \sum_I X_i, X, \Gamma' \; ; \; \Delta) \\ \hline\hline
\prod_I \mathbf{X}(\Gamma, X_i, X, \Gamma' \; ; \; \Delta) \\ \hline\hline
\prod_I \mathbf{X}(\Gamma, X_i \otimes X, \Gamma' \; ; \; \Delta) \\ \hline\hline
\mathbf{X}(\Gamma, \sum_I (X_i \otimes X), \Gamma' \; ; \; \Delta)
\end{array}$$

[$Y_* \otimes B = (Y \mathbin{\langle\!\otimes} B)_*$]

We want a bijection between polyarrows $\Gamma, Y_* \otimes B, \Gamma' \vdash \Delta$ and polyarrows $\Gamma, (Y \mathbin{\langle\!\otimes} B)_*, \Gamma' \vdash \Delta$, where $\vdash$ may be any appropriate type of polyarrow (in $\mathbf{X}_o, \mathbf{X}_p, \widehat{\mathbf{X}}$), depending on what, if anything, is in focus. In this case, we need to do an induction on the last step in a derivation of a sequent $\Gamma, (Y \mathbin{\langle\!\otimes} B)_*, \Gamma' \vdash \Delta$, to construct a corresponding sequent $\Gamma, Y_* \otimes B, \Gamma' \vdash \Delta$. If that last step doesn't involve $(Y \mathbin{\langle\!\otimes} B)_*$, then there is no problem — merely mimic that step. But if the last step does involve $(Y \mathbin{\langle\!\otimes} B)_*$, then its form must be

$$\begin{array}{c}
\pi \\ \hline
\Gamma \,/\, Y \mathbin{\langle\!\otimes} B \,\backslash\, \Gamma' \vdash_p \Delta \\ \hline
\Gamma, (Y \mathbin{\langle\!\otimes} B)_*, \Gamma' \mathrel{\underset{o\;p}{\mapsto}} \Delta
\end{array}$$

Then the induced derivation is constructed thus (where we indicate a standard bijection given by representability *via* a double horizontal line):

$$\begin{array}{c}
\pi \\ \hline
\Gamma \,/\, Y \mathbin{\langle\!\otimes} B \,\backslash\, \Gamma' \vdash_p \Delta \\ \hline\hline
\Gamma \,/\, Y \,\backslash\, B, \Gamma' \vdash_p \Delta \\ \hline
\Gamma, Y_*, B, \Gamma' \mathrel{\underset{o\;p}{\mapsto}} \Delta \\ \hline
\Gamma, Y_* \otimes B, \Gamma' \mathrel{\underset{o\;p}{\mapsto}} \Delta
\end{array}$$

For the reverse direction, if the last two steps of a derivation of $\Gamma, Y_* \otimes B, \Gamma' \vdash \Delta$ have Y_*, B in focus, in essence we reverse the step above:

$$\begin{array}{c}
\pi \\ \hline
\Gamma \,/\, Y \,\backslash\, B, \Gamma' \vdash_p \Delta \\ \hline
\Gamma, Y_*, B, \Gamma' \mathrel{\underset{o\;p}{\mapsto}} \Delta \\ \hline
\Gamma, Y_* \otimes B, \Gamma' \mathrel{\underset{o\;p}{\mapsto}} \Delta
\end{array}
\quad \mapsto \quad
\begin{array}{c}
\pi \\ \hline
\Gamma \,/\, Y \,\backslash\, B, \Gamma' \vdash_p \Delta \\ \hline
\Gamma \,/\, Y \mathbin{\langle\!\otimes} B \,\backslash\, \Gamma' \vdash_p \Delta \\ \hline
\Gamma, (Y \mathbin{\langle\!\otimes} B)_*, \Gamma' \mathrel{\underset{o\;p}{\mapsto}} \Delta
\end{array}$$

If, on the other hand, the penultimate step does not involve Y_*, B, then we can move the induction up a step, for example as follows.

$$\begin{array}{c}
\left\{ \begin{array}{c} \pi_i \\ \hline \Gamma, Y_*, B, \Gamma', X_i, \Gamma'' \mathrel{\underset{o\;p}{\mapsto}} \Delta \end{array} \right\}_I \\ \hline
\Gamma, Y_*, B, \Gamma', \sum_I X_i, \Gamma'' \mathrel{\underset{o\;p}{\mapsto}} \Delta \\ \hline
\Gamma, Y_* \otimes B, \Gamma', \sum_I X_i, \Gamma'' \mathrel{\underset{o\;p}{\mapsto}} \Delta
\end{array}
\quad \equiv \quad
\begin{array}{c}
\left\{ \begin{array}{c} \pi_i \\ \hline \Gamma, Y_*, B, \Gamma', X_i, \Gamma'' \mathrel{\underset{o\;p}{\mapsto}} \Delta \\ \hline \Gamma, Y_* \otimes B, \Gamma', X_i, \Gamma'' \mathrel{\underset{o\;p}{\mapsto}} \Delta \end{array} \right\}_I \\ \hline
\Gamma, Y_* \otimes B, \Gamma', \sum_I X_i, \Gamma'' \mathrel{\underset{o\;p}{\mapsto}} \Delta
\end{array}$$

which we map (inductively) to

$$\dfrac{\left\{\dfrac{\widehat{\pi}_i}{\Gamma, (Y \mathbin{(\!\otimes} B)_*, \Gamma', X_i, \Gamma'' \underset{o\ p}{\mapsto} \Delta}\right\}_I}{\Gamma, (Y \mathbin{(\!\otimes} B)_*, \Gamma', \sum_I X_i, \Gamma'' \underset{o\ p}{\mapsto} \Delta}$$

where $\widehat{\pi}_i$ is the derivation assigned to π_i.

Since B is atomic, the only proof step in which $Y_* \otimes B$ is active is unfocusing the Y_*. The essence of the process here is to trace back through the derivation until we have the two steps together, when we can proceed as we sketched above. We shall do essentially the same thing for our next case.

[$X_* \otimes Y_* = (X \mathbin{(\!\otimes} Y_*) \times (X_* \mathbin{\otimes\!)} Y)$]

We permute steps of the derivations so that those involving X_*, Y_* are together, and then use the correspondence (or the similar one for the other projection) below.

$$\dfrac{\dfrac{\dfrac{\pi}{\Gamma \,/\, X \setminus Y_*, \Gamma' \vdash_p \Delta}}{\Gamma, X_*, Y_*, \Gamma' \underset{o\ p}{\mapsto} \Delta}}{\Gamma, X_* \otimes Y_*, \Gamma' \underset{o\ p}{\mapsto} \Delta} \quad \longleftrightarrow \quad \dfrac{\dfrac{\dfrac{\pi}{\Gamma \,/\, X \setminus Y_*, \Gamma' \vdash_p \Delta}}{\Gamma, X \mathbin{(\!\otimes} Y_*, \Gamma' \underset{o\ p}{\mapsto} \Delta}}{\Gamma, (X \mathbin{(\!\otimes} Y_*) \times (X_* \mathbin{\otimes\!)} Y), \Gamma' \underset{o\ p}{\mapsto} \Delta}$$

[$\bigwedge_I X_i \otimes Y = \bigwedge_I (X_i \otimes Y)$]

$$\begin{array}{c} \mathbf{X}(\Gamma, \bigwedge_I X_i \otimes Y, \Gamma' \ ;\ \Delta) \\ \hline\hline \mathbf{X}(\Gamma, \bigwedge_I X_i, Y, \Gamma' \ ;\ \Delta) \\ \hline\hline \sum_I \mathbf{X}(\Gamma, X_i, Y, \Gamma' \ ;\ \Delta) \\ \hline\hline \sum_I \mathbf{X}(\Gamma, X_i \otimes Y, \Gamma' \ ;\ \Delta) \\ \hline\hline \mathbf{X}(\Gamma, \bigwedge_I (X_i \otimes Y), \Gamma' \ ;\ \Delta) \end{array}$$

[$X \mathbin{\otimes\!)} \bigvee_I X_i = \bigvee_I (X \mathbin{\otimes\!)} X_i)$]

$$\begin{array}{c} \mathbf{X}_p(\Gamma \,/\, X \mathbin{\otimes\!)} \bigvee_I X_i \setminus \Gamma' \ ;\ \Delta) \\ \hline\hline \mathbf{X}_p(\Gamma, X \,/ \bigvee_I X_i \setminus \Gamma' \ ;\ \Delta) \\ \hline\hline \prod_I \mathbf{X}_p(\Gamma, X \,/\, X_i \setminus \Gamma' \ ;\ \Delta) \\ \hline\hline \prod_I \mathbf{X}_p(\Gamma \,/\, \mathbf{X} \mathbin{\otimes\!)} X_i \setminus \Gamma' \ ;\ \Delta) \\ \hline\hline \mathbf{X}_p(\Gamma \,/ \bigvee_I (X \mathbin{\otimes\!)} X_i) \setminus \Gamma' \ ;\ \Delta) \end{array}$$

[$Y' \mathbin{\otimes\!)} Y^* = (Y' \otimes Y)^*$]

$$\begin{array}{c} \mathbf{X}_p(\Gamma \,/\, Y' \mathbin{\otimes\!)} Y^* \setminus \Gamma' \ ;\ \Delta) \\ \hline\hline \mathbf{X}_p(\Gamma, Y' \,/\, Y^* \setminus \Gamma' \ ;\ \Delta) \\ \hline\hline \widehat{\mathbf{X}}(\Gamma, Y', Y, \Gamma' \ ;\ \Delta) \\ \hline\hline \widehat{\mathbf{X}}(\Gamma, Y' \otimes Y, \Gamma' \ ;\ \Delta) \\ \hline\hline \mathbf{X}_p(\Gamma \,/ (Y' \otimes Y)^* \setminus \Gamma' \ ;\ \Delta) \end{array}$$

[$\bigwedge_I X_i \otimes\!\rangle\, Y = \prod_I (X_i \otimes\!\rangle\, Y)$]

$$\begin{array}{c}
\mathbf{X}_p(\Gamma \,/\, \bigwedge_I X_i \otimes\!\rangle\, Y \setminus \Gamma' \;;\; \Delta \\
\hline\hline
\mathbf{X}_p(\Gamma, \bigwedge_I X_i \,/\, Y \setminus \Gamma' \;;\; \Delta \\
\hline\hline
\sum_I \mathbf{X}_p(\Gamma, X_i \,/\, Y \setminus \Gamma' \;;\; \Delta \\
\hline\hline
\sum_I \mathbf{X}_p(\Gamma \,/\, X_i \otimes\!\rangle\, Y \setminus \Gamma' \;;\; \Delta \\
\hline\hline
\mathbf{X}_p(\Gamma \,/\, \prod_I (X_i \otimes\!\rangle\, Y) \setminus \Gamma' \;;\; \Delta
\end{array}$$

[$Y_* \otimes\!\rangle\, B = \prod_\emptyset$] Since B is atomic, and since there is no unfocusing in $\mathbf{X}$, we see that the following homsets are empty.

$$\mathbf{X}_p(\Gamma \,/\, Y_* \otimes\!\rangle\, B \setminus \Gamma' \;;\; \Delta) \simeq \mathbf{X}_p(\Gamma, Y_* \,/\, B \setminus \Gamma' \;;\; \Delta) \simeq \emptyset \simeq \mathbf{X}_p(\Gamma, Y_* \,/\, \prod_\emptyset \setminus \Gamma' \;;\; \Delta)$$

Note that $\prod_\emptyset$ is the fore-terminal object (the unit for the fore products).

[$\sum_I X_i \otimes \prod_J Y_j = \bigvee_I \prod_J (X_i \otimes\!\rangle\, Y_j)$] We have already illustrated this bijection, in the proof of Lemma 6.8.1.

∎

7. Multiplicative and additive structure on AJ games

Our prime example of a polarized game polycategory is representable; in fact, the multiplicative structure on AJ games was a primary inspiration for the general treatment of representability. It seems useful, then, to give an exposition of the representing structure for AJ games.

7.1. Multiplicative structure. We begin by defining various binary ("multiplicative") operations on the combinatorial games of section 1.1. Suppose we have games

$$O = (b_j \colon P_j \mid j \in J),\; O' = (b'_j \colon P'_j \mid j \in J'),\; P = \{a_i \colon O_i \mid i \in I\},\; P' = \{a'_i \colon O'_i \mid i \in I'\}$$

[$O \otimes\!\rangle\, P$] This operation takes an opponent game O and a player game P and produces a player game. The operation is defined recursively by:

$$O \otimes\!\rangle\, P = \{a_i \colon O \otimes O_i \mid i \in I\}$$

where the tensor operation $\otimes$ is defined below. This operation behaves rather like the tensor, apart from its typing. Notice that the "direction" of the operation points to the game whose type is inherited by $O \otimes\!\rangle\, P$.

[$P \,\langle\!\otimes\, O$] This operation is a simple left-right dual of the preceding one, taking a player game P and an opponent game O, producing a player game. The operation is defined recursively by:

$$P \,\langle\!\otimes\, O = \{a_i \colon O_i \otimes O \mid i \in I\}$$

[$O \mathbin{\langle\!\oplus} P$] This operation takes an opponent game O and a player game P and produces an opponent game. The operation is defined recursively by:

$$O \mathbin{\langle\!\oplus} P = (b_j\colon P'_j \oplus P \mid j \in J)$$

where the operation par $\oplus$ is defined below. Again, this behaves like a variant of $\oplus$, but with different typing. Again, the symbol points to the game whose type is inherited by the compound game. This convention should help the reader (not to mention the authors!) keep track of the "opponent-player" typing.

[$P \mathbin{\oplus\!\rangle} O$] This operation is the left-right dual, taking a player game P and an opponent game O, producing an opponent game. The operation is defined recursively by:

$$P \mathbin{\oplus\!\rangle} O = (b_j\colon P \oplus P'_j \mid j \in J)$$

[$O \otimes O'$] This operation takes in two opponent games and produces an opponent game. It is defined recursively by:

$$O \otimes O' = (b_j\colon P_j \mathbin{\langle\!\otimes} O', b'_k\colon O \mathbin{\otimes\!\rangle} P'_k \mid j \in J, k \in J')$$

[$P \oplus P'$] This operation takes in two player games and produces a player game. It is defined recursively by:

$$P \oplus P' = \{a_i\colon P' \mathbin{\oplus\!\rangle} O_i, a'_k\colon O'_k \mathbin{\langle\!\oplus} P \mid i \in I, k \in I'\}$$

The following two aft additive operations are also of interest; we add them here for reference.

[$O \wedge O'$] This operation takes two opponent games and produces an opponent game by "merging" the games together.

$$O \wedge O' = (b_j\colon P_j, b'_k\colon P'_k \mid j \in J, k \in J')$$

[$P \vee P'$] This operation takes two player games and produces a player game by merging them.

$$P \vee P' = \{a_i\colon O_i, a'_k\colon O'_j \mid i \in I, k \in I'\}$$

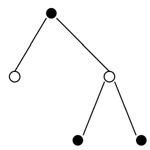
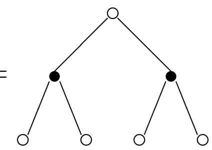
Then for the record we state the following proposition (the proof is essentially obvious, and is implicit in Abramsky's papers):

7.1.1. Proposition. *With the operations as defined above, AJ games are a linear polarized game category.*

7.1.2. Remark. There are a number of simple identities that hold of these operations; we list (without proof) several for reference.

$$\begin{array}{rcl}
P \otimes P' &=& \overline{\overline{P} \oplus \overline{P'}} \\
O \wedge O' &=& \overline{\overline{O} \vee \overline{O'}} \\
O \otimes O' &=& O' \otimes O \\
O \otimes \mathbf{1} &=& O \\
O \mathbin{\otimes\!\rangle} \mathbf{0} &=& \mathbf{0} \\
\mathbf{1} \mathbin{\otimes\!\rangle} P &=& P \\
(O \otimes O') \mathbin{\otimes\!\rangle} P &=& O \mathbin{\otimes\!\rangle} (O' \mathbin{\otimes\!\rangle} P) \\
O \mathbin{\otimes\!\rangle} (O_1 \sqcup O_2) &=& (O \otimes O_1) \sqcup (O \otimes O_2)
\end{array}
\qquad
\begin{array}{rcl}
P \sqcap P' &=& \overline{\overline{P} \sqcup \overline{P'}} \\
O \mathbin{\otimes\!\rangle} P &=& \overline{\overline{P} \mathbin{\langle\!\oplus} \overline{O}} \\
P \oplus P' &=& P' \oplus P \\
\mathbf{0} \oplus P &=& P \\
\mathbf{1} \mathbin{\langle\!\oplus} P &=& \mathbf{1} \\
O \mathbin{\langle\!\oplus} \mathbf{0} &=& O \\
O \mathbin{\langle\!\oplus} (P \oplus P') &=& (O \mathbin{\langle\!\oplus} P) \mathbin{\langle\!\oplus} P' \\
(P_1 \sqcap P_2) \mathbin{\langle\!\oplus} P &=& (P_1 \oplus P) \sqcap (P_2 \oplus P)
\end{array}$$

7.1.3. Remark. A useful technique in describing games is to use multiplicities. Intuitively if a subgame occurs multiple times, rather than write the game multiple times one can simply multiply it by the number of times it occurs. For example, $P = \{1\colon (\ ), (2\colon \{\ \})\}$ and $Q = (2\colon \{2\colon (\ )\})$ would be the following two games.

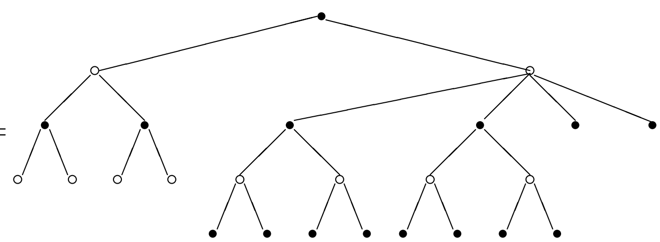
$P =$  $Q =$ 

We shall often drop the labelling to facilitate calculations.

Then we can calculate (using the identities above to help when possible)

$$\begin{aligned}
Q \otimes P &= \{1\colon Q \otimes (\ ), 1\colon Q \otimes (2\colon \{\ \})\} \\
&= \{1\colon Q, 1\colon (2\colon (2\colon \{\ \}) \otimes \{2\colon (\ )\}, 2\colon Q \otimes \{\ \})\} \\
&= \{1\colon Q, 1\colon (2\colon \{2\colon (2\colon \{\ \}) \otimes (\ )\}, 2\colon \{\ \})\} \\
&= \{1\colon Q, 1\colon (2\colon \{2\colon (2\colon \{\ \})\}, 2\colon \{\ \})\}
\end{aligned}$$

which graphically is

$Q \otimes P =$ 

It is worth noting that multiplicities are actually more than just a notational convenience: they are actually a tensor and a cotensor in the sense of enriched categories [K82] and are an indication that the games we are introducing are finite set-enriched. We hope to investigate this further later, discussing polarized categories enriched over a monoidal closed category. (This idea has antecedents in Joyal's "money games" [J95].)

7.2. The structure of finite AJ games. We start by observing that the opponent category is closed with respect to $\otimes$. To this end, we define an internal hom for the opponent category *via* $A \multimap B = \overline{A} \oplus B$. It is then possible (see [H97] for example) to verify that:

7.2.1. Proposition. *The opponent category is a closed monoidal category with respect to* $\otimes$ *and* $\multimap$ *as defined above.*

Note that by duality the player category has an internal co-hom with respect to $\oplus$.

Furthermore, we can show that combinatorial basic polarized games (that is, finite Abramsky–Jagadeesan games) form the initial polarized category with polarized products and coproducts.

7.2.2. Proposition. *The following data is the initial model for the basic game logic:*

- *Player games with maps* $\tau: P_1 \longrightarrow P_2$ *which are strategies in* $\overline{P_2} \otimes P_1$,
- *Opponent games with maps* $\tau': O_1 \longrightarrow O_2$ *where* τ' *is a strategy in* $O_2 \oplus \overline{O_1}$,
- *Module maps* $\delta: O \longrightarrow P$ *where* δ *is a counter-strategy in* $\overline{O} \oplus P$, *or equivalently, a counter-strategy in* $O \otimes \overline{P}$.

Proof. We prove this by induction, by showing that the deduction rules preserve this characterization of each of the three types of sequent. To start with, the only atomic games are $\mathbf{1}$ and $\mathbf{0}$ — the identity $\mathbf{1} \vdash_o \mathbf{1}$ is supposed to correspond to a strategy for $\mathbf{1} \oplus \mathbf{0} = \mathbf{1}$, which is the empty counter-strategy. Likewise, the identity $\mathbf{0} \vdash_p \mathbf{0}$ corresponds to the empty counter-strategy strategy for $\mathbf{1} \otimes \mathbf{0} = \mathbf{0}$.

Consider the injection rule: we suppose we have $f: O \vdash_o O_k$ given by a strategy for $O_k \oplus \overline{O}$, where $k \in I$ and O_k is a member of a family of opponent games O_i, $i \in I$. We want to see that the induced $\overrightarrow{b_k} \cdot f: Q \underset{o\ p}{\vdash\!\!\rightarrow} \bigsqcup_{i \in I} O_i$ is given by a counter-strategy for $\overline{O} \oplus P$, where $P = \bigsqcup_{i \in I} = \{O_i \mid i \in I\}$. But $\overline{O} \oplus P = \{\overline{P_j} \oplus P, O_i \oplus \overline{O} \mid j \in J, i \in I\}$, where $O = (P_j \mid j \in J)$, and a counter-strategy for that is canonically given by choosing the given strategy for $O_k \oplus \overline{O}$. The projection rule is handled dually.

For the cotuple rule, we suppose we have given a family $f_i: O_i \underset{o\ p}{\vdash\!\!\rightarrow} P'$ $(i \in I)$, each of which is given by a counter-strategy for $O_i \otimes \overline{P'}$. We want to see that the induced $\{f_i\}_{i \in I}: \bigsqcup_{i \in I} O_i \vdash_p P'$ is given by a strategy for $\overline{P'} \otimes P$, where $P = \bigsqcup_{i \in I} O_i = \{O_i \mid i \in I\}$. But $\overline{P'} \otimes P = \{\overline{P'} \otimes O_i \mid i \in I\}$, and so a counter-strategy for this amounts to having a counter-strategy for each $\overline{P'} \otimes O_i$, which we have by assumption. The tuple rule is dual.

We do not have to worry about the cut rules, since by cut elimination, we may represent each morphism in the free model by a cut-free proof. So this completes the proof. ∎

Part IV Extending game structures

8. Depolarization

There is one major difference between our finite AJ games and the actual game categories constructed by Abramsky and Jagadeesan: their categories were more than linear, they were $*$-autonomous, and moreover they were not explicitly polarized. We can recover the actual category constructed by them *via* a construction called "depolarization". Although in effect Abramsky and Jagadeesan used the construction at the level of linear polarized categories, the construction is more general and best understood at the level of polarized polycategories.

8.0.3. Definition. *Given a polarized polycategory* $\mathbf{X}$*, we define its depolarization* $\mathsf{dePol}(\mathbf{X})$ *as the polycategory* $\mathbf{Y}$ *whose objects are pairs* $\langle Q, P\rangle$*, where* $Q \in \mathbf{X}_o$ *and* $P \in \mathbf{X}_p$*, and whose polyarrows*

$$\langle Q_1, P_1\rangle, \ldots, \langle Q_m, P_m\rangle \longrightarrow \langle Q'_1, P'_1\rangle, \ldots, \langle Q'_n, P'_n\rangle$$

correspond to $n+m$*-tuples of polarized polyarrows of the following sorts (for* $1 \le i \le m$ *and* $1 \le j \le n$*)*

$$\begin{array}{ll} Q_1, \ldots, Q_{i-1} \,/\, P_i \,\backslash\, Q_{i+1}, \ldots, Q_m \longrightarrow P'_1, \ldots, P'_n & \text{in } \mathbf{X}_p \\ Q_1, \ldots, Q_m \longrightarrow P'_1, \ldots, P'_{j-1} \,/\, Q'_j \,\backslash\, P'_{j+1}, \ldots, P'_n & \text{in } \mathbf{X}_o, \text{ and} \end{array}$$

To define the cut of such polyarrows, one merely performs all the $k+l-2$ *appropriate cuts in the sets of polyarrows in* $\mathbf{X}$ *(*k, l *being the total number of formulas in the two polyarrows being cut).*

To illustrate cut in $\mathsf{dePol}(\mathbf{X})$, consider the cut $f\,;g$ of:

$$(Q_1, P_1), (Q_2, P_2) \xrightarrow{f} (Q'_1, P'_1), (Q, P), (Q'_2, P'_2)$$

with

$$(Q_3, P_3), (Q, P), (Q_4, P_4) \xrightarrow{g} (Q'_3, P'_3), (Q'_4, P'_4)$$

These polyarrows consist of the following polyarrows in $\mathbf{X}$:

$$\begin{array}{rcl@{\qquad}rcl} /\, P_1 \,\backslash\, Q_2 & \xrightarrow{f_1} & P'_1, P, P'_2 & /\, P_3 \,\backslash\, Q, Q_4 & \xrightarrow{g_1} & P'_3, P'_4 \\ Q_1 \,/\, P_2 \,\backslash & \xrightarrow{f_2} & P'_1, P, P'_2 & Q_3 \,/\, P \,\backslash\, Q_4 & \xrightarrow{g_2} & P'_3, P'_4 \\ Q_1, Q_2 & \xrightarrow{f_3} & /\, Q'_1 \,\backslash\, P, P'_2 & Q_3, Q \,/\, P_4 \,\backslash & \xrightarrow{g_3} & P'_3, P'_4 \\ Q_1, Q_2 & \xrightarrow{f_4} & P'_1 \,/\, Q \,\backslash\, P'_2 & Q_3, Q, Q_4 & \xrightarrow{g_4} & /\, Q'_3 \,\backslash\, P'_4 \\ Q_1, Q_2 & \xrightarrow{f_5} & P'_1, P \,/\, Q'_2 \,\backslash & Q_3, Q, Q_4 & \xrightarrow{g_5} & P'_3 \,/\, Q'_4 \,\backslash \end{array}$$

We cut the single occurrence of Q in focus with the four instances of Q out of focus and the single occurrence of P in focus with the four instances of P out of focus to get the polyarrow $f\,;g$ in $\mathsf{dePol}(\mathbf{X})$.

$$(Q_3, P_3), (Q_1, P_1), (Q_2, P_2), (Q_4, P_4) \xrightarrow{f;g} (Q'_1, P'_1), (Q'_3, P'_3), (Q'_4, P'_4), (Q'_2, P'_2)$$

$$
\begin{array}{rcl}
/\, P_3 \,\backslash\, Q_1, Q_2, Q_4 & \xrightarrow{f_4;g_1} & P_1', P_3', P_4', P_2' \\
Q_3 \,/\, P_1 \,\backslash\, Q_2, Q_4 & \xrightarrow{f_1;g_2} & P_1', P_3', P_4', P_2' \\
Q_3, Q_1 \,/\, P_2 \,\backslash\, Q_4 & \xrightarrow{f_2;g_2} & P_1', P_3', P_4', P_2' \\
Q_3, Q_1, Q_2 \,/\, P_4 \,\backslash & \xrightarrow{f_4;g_3} & P_1', P_3', P_4', P_2' \\
Q_3, Q_1, Q_2, Q_4 & \xrightarrow{f_3;g_2} & /\, Q_1' \,\backslash\, P_3', P_4', P_2' \\
Q_3, Q_1, Q_2, Q_4 & \xrightarrow{f_4;g_4} & P_1' \,/\, Q_3' \,\backslash\, P_4', P_2' \\
Q_3, Q_1, Q_2, Q_4 & \xrightarrow{f_4;g_5} & P_1', P_3' \,/\, Q_4' \,\backslash\, P_2' \\
Q_3, Q_1, Q_2, Q_4 & \xrightarrow{f_5;g_2} & P_1', P_3', P_4', /\, Q_2' \,\backslash
\end{array}
$$

Note that ordinary arrows in $\mathsf{dePol}(\mathbf{X})$ are pairs $\langle \sigma, \tau \rangle : \langle Q, P \rangle \longrightarrow \langle Q', P' \rangle$ consisting of a morphism $\sigma : Q \longrightarrow Q'$ of $\mathbf{X}_o$ and a morphism $\tau : P \longrightarrow P'$ of $\mathbf{X}_p$, with the obvious composition.

8.0.4. Proposition. $\mathsf{dePol}(\mathbf{X})$ *is a polycategory.*

Proof. The validity of the polycategory axioms obviously lifts to $\mathsf{dePol}(\mathbf{X})$ from $\mathbf{X}$. ■

This construction is somewhat reminiscent of a Chu-style construction, considering that we can regard $\mathbf{X}_p$ as "dual" to $\mathbf{X}_o$. But note that it is not a Chu construction, in that it does not give us negation if the original structure has no negation. This construction recaptures the "original" Abramsky–Jagadeesan games from their polarized variety.

8.1. Representability of depolarization. The process of depolarization preserves representability, in the following sense, when the underlying (representable) polarized polycategory has aft additives.

8.1.1. Theorem. *If $\mathbf{X}$ is a linear polarized category with aft additive structure, then $\mathsf{dePol}(\mathbf{X})$ is representable. That is, there are tensor and par connectives which 'make it' a linearly distributive category. If in addition $\mathbf{X}$ has negation, then $\mathsf{dePol}(\mathbf{X})$ 'is' a $*$-autonomous category.*

Proof. Tensor and par on $\mathsf{dePol}(\mathbf{X})$ are constructed as follows.

$$
\begin{array}{rcl}
\langle Q, P \rangle \boxtimes \langle Q', P' \rangle & = & \langle Q \otimes Q', Q \otimes\!\rangle\, P' \vee P \,\langle\!\otimes Q' \rangle \\
\langle Q, P \rangle \boxplus \langle Q', P' \rangle & = & \langle Q \,\langle\!\oplus P' \wedge P \oplus\!\rangle\, Q', P \oplus P' \rangle
\end{array}
$$

The $\boxtimes$ unit is $\langle \top, 0 \rangle$, and dually, the $\boxplus$ unit is $\langle 1, \bot \rangle$.

We must now show that there are universal polyarrows corresponding to these connectives. We shall do this for the tensor, the par being dual. The universal polyarrow

$$
\langle Q, P \rangle, \langle Q', P' \rangle \longrightarrow \langle Q \otimes Q', Q \otimes\!\rangle\, P' \vee P \,\langle\!\otimes Q' \rangle
$$

by definition must be given by polarized polyarrows as follows:

$$
\begin{array}{rcl}
Q, Q' & \longrightarrow & /\, Q \otimes Q' \,\backslash \\
Q \,/\, P' \,\backslash & \longrightarrow & Q \otimes\!\rangle\, P' \vee P \,\langle\!\otimes Q' \\
/\, P \,\backslash\, Q' & \longrightarrow & Q \otimes\!\rangle\, P' \vee P \,\langle\!\otimes Q'
\end{array}
$$

The first of these is given by the representability of $\mathbf{X}$; the other two are the two injections combined with the polarized polyarrows $Q\,/\,P'\,\backslash \longrightarrow Q \mathbin{\otimes\!\!\rangle} P'$ and $/\,P\,\backslash\,Q' \longrightarrow P \mathbin{\langle\!\!\otimes} Q'$, also given by representability. We must show that cutting with this universal polyarrow induces a bijection

$$\frac{\Gamma, \langle Q, P\rangle, \langle Q', P'\rangle, \Gamma' \xrightarrow{f} \Delta}{\Gamma, \langle Q \otimes Q', Q \mathbin{\otimes\!\!\rangle} P' \vee P \mathbin{\langle\!\!\otimes} Q'\rangle, \Gamma' \xrightarrow{\hat{f}} \Delta}$$

where Γ, Γ', Δ represent strings of pairs $\langle Q_i, P_i\rangle$. Given a polyarrow f as above, we must construct the polyarrow $\hat{f}$ corresponding to it, the reverse association given by cut. Now, f corresponds to a number of polarized polyarrows in $\mathbf{X}$; some are $\mathbf{X}_o$ arrows, which will have the pair Q, Q' on the left. By representability of $\mathbf{X}$, such arrows bijectively correspond to polarized polyarrows with the Q, Q' replaced by $Q \otimes Q'$, part of the data needed for $\hat{f}$. Other polarized polyarrows comprising f will be $\mathbf{X}_p$ arrows with neither P nor P' in the active position on the left; such polarized polyarrows will have Q, Q' in the context, and so correspond to similar polarized polyarrows with $Q \otimes Q'$ in the context. Finally, there will be polarized polyarrows with either P or P' in the active position on the left; such polarized polyarrows will have the configuration $\ldots/\,P\,\backslash\,Q'\ldots$ or $\ldots Q\,/\,P'\,\backslash\ldots$ on the left, and so correspond (by representability) to polarized polyarrows with the configuration (respectively) $\ldots/\,P \mathbin{\langle\!\!\otimes} Q'\,\backslash\ldots$ or $\ldots/\,Q \mathbin{\otimes\!\!\rangle} P'\,\backslash\ldots$ on the left. *Via* an appropriate injection, either of these will induce a polarized polyarrow with the configuration $\ldots/\,Q \mathbin{\otimes\!\!\rangle} P' \vee P \mathbin{\langle\!\!\otimes} Q'\,\backslash\ldots$ on the left. In this way we have constructed the data needed for $\hat{f}$. It should be clear from the construction of the universal polyarrow that these associations are bijective, as required. Naturality is inherited from $\mathbf{X}$.

To see that $\langle\top, 0\rangle$ is the tensor unit, the argument is similar. The main observations to make are that there is a unique $\ldots/\,0\,\backslash\ldots \longrightarrow \ldots$ so that will have no effect, and the bijection between $\ldots\top\ldots \longrightarrow \ldots$ and $\ldots\quad\ldots \longrightarrow \ldots$ comes from representability of the tensor unit in $\mathbf{X}$.

Finally, if $\mathbf{X}$ has (polarized) negation, then we can define negation in $\mathsf{dePol}(\mathbf{X})$: $\langle Q, P\rangle^{\perp} = \langle P^{\perp}, Q^{\perp}\rangle$, and dually on the left. The (left) τ for this is

$$\langle\top, 0\rangle \longrightarrow \langle Q, P\rangle \boxplus \langle {}^{\perp}P, {}^{\perp}Q\rangle = \langle Q \mathbin{\langle\!\!\oplus} {}^{\perp}Q \wedge P \mathbin{\oplus\!\!\rangle} {}^{\perp}P, P \oplus {}^{\perp}Q\rangle$$

the (right) τ and the γ's are dual. The second component $0 \longrightarrow B \oplus {}^{\perp}Q$ exists since 0 is initial. The first component is derived canonically, using the left τ's from $\mathbf{X}$. It is straightforward to show this inherits the necessary coherence conditions from $\mathbf{X}$, and so $\mathsf{dePol}(\mathbf{X})$ is a (non-commutative) $*$-autonomous category. ∎

To get a feel for the depolarized linear category, let us consider some simple examples. First it might be reassuring for the reader to note that the definition of the units is forced by the following simple calculation.

$$\begin{array}{rcl}\langle\top, 0\rangle \boxtimes \langle Q, P\rangle & \cong & \langle\top \otimes Q, \top \mathbin{\otimes\!\!\rangle} P \vee 0 \mathbin{\langle\!\!\otimes} Q\rangle \\ & \cong & \langle Q, P \vee 0\rangle \;\cong\; \langle Q, P\rangle\end{array}$$

where we have used the linearity of the (cartesian) sum in the equation $0 \mathbin{\langle\!\!\otimes} Q \cong 0$.

Next, let us illustrate the associativity of $\boxtimes$ and the linear distribution. We shall do this by constructing the necessary derivations, but first, let's calculate some expressions.

$$\begin{array}{rl}
\multicolumn{2}{l}{(\langle Q_1, P_1\rangle \boxtimes \langle Q_2, P_2\rangle) \boxtimes \langle Q_3, P_3\rangle} \\
= & \langle Q_1 \otimes Q_2, Q_1 \otimes\!\rangle\, P_2 \vee P_1 \,\langle\!\otimes Q_2\rangle \boxtimes \langle Q_3, P_3\rangle \\
= & \langle Q_1 \otimes Q_2 \otimes Q_3, (Q_1 \otimes Q_2) \otimes\!\rangle\, P_3 \vee (Q_1 \otimes\!\rangle\, P_2 \vee P_1 \,\langle\!\otimes Q_2) \,\langle\!\otimes Q_3\rangle \\
\multicolumn{2}{l}{\langle Q_1, P_1\rangle \boxtimes (\langle Q_2, P_2\rangle \boxtimes \langle Q_3, P_3\rangle)} \\
= & \langle Q_1, P_1\rangle \boxtimes \langle Q_2 \otimes Q_3, Q_2 \otimes\!\rangle\, P_3 \vee P_2 \,\langle\!\otimes Q_3\rangle \\
= & \langle Q_1 \otimes Q_2 \otimes Q_3, Q_1 \otimes\!\rangle\, (Q_2 \otimes\!\rangle\, P_3 \vee P_2 \,\langle\!\otimes Q_3) \vee P_1 \,\langle\!\otimes (Q_2 \otimes Q_3)\rangle
\end{array}$$

To verify these are isomorphic, it suffices to note that $Q_1 \otimes\!\rangle\, (Q_2 \otimes\!\rangle\, P) \cong (Q_1 \otimes Q_2) \otimes\!\rangle\, P$ and $Q \otimes\!\rangle\, (P_1 \vee P_2) \cong Q \otimes\!\rangle\, P_1 \vee Q \otimes\!\rangle\, P_2$ (polarized associativity and linear (cartesian) sums in $\mathbf{X}$).

Next, the linear distribution; first we calculate the terms involved.

$$\begin{array}{rl}
\multicolumn{2}{l}{\langle Q_1, P_1\rangle \boxtimes (\langle Q_2, P_2\rangle \boxplus \langle Q_3, P_3\rangle)} \\
= & \langle Q_1, P_1\rangle \boxtimes \langle Q_2 \,\langle\!\oplus P_3 \wedge P_2 \oplus\!\rangle\, Q_3, P_2 \oplus P_3\rangle \\
= & \langle Q_1 \otimes (Q_2 \,\langle\!\oplus P_3 \wedge P_2 \oplus\!\rangle\, Q_3), Q_1 \otimes\!\rangle\, P_2 \oplus P_3 \vee P_1 \,\langle\!\otimes (Q_2 \,\langle\!\oplus P_3 \wedge P_2 \oplus\!\rangle\, Q_3)\rangle \\
\text{and} & (\langle Q_1, P_1\rangle \boxtimes \langle Q_2, P_2\rangle) \boxplus \langle Q_3, P_3\rangle \\
= & \langle Q_1 \otimes Q_2, Q_1 \otimes\!\rangle\, P_2 \vee P_1 \,\langle\!\otimes Q_2\rangle \boxplus \langle Q_3, P_3\rangle \\
= & \langle (Q_1 \otimes Q_2) \,\langle\!\oplus P_3 \wedge (Q_1 \otimes\!\rangle\, P_2 \vee P_1 \,\langle\!\otimes Q_2) \oplus\!\rangle\, Q_3, (Q_1 \otimes\!\rangle\, P_2 \vee P_1 \,\langle\!\otimes Q_2) \oplus P_3\rangle
\end{array}$$

By symmetry, this is essentially induced by

$$Q_1 \otimes (Q_2 \,\langle\!\oplus P_3 \wedge P_2 \oplus\!\rangle\, Q_3) \longrightarrow (Q_1 \otimes Q_2) \,\langle\!\oplus P_3 \wedge (Q_1 \otimes\!\rangle\, P_2 \vee P_1 \,\langle\!\otimes Q_2) \oplus\!\rangle\, Q_3$$

which we have illustrated earlier, section 6.3. So we can see the structure of a linearly distributive category. (Note this doesn't need further proof, since we know that a representable polycategory *is* a linearly distributive category.)

8.2. Additives and depolarization. We remark that if $\mathbf{X}$ has (fore and aft) additives, then $\mathsf{dePol}(\mathbf{X})$ has (ordinary) sums and products:

8.2.1. Proposition. *If $\mathbf{X}$ is a linear polarized category with fore and aft additive structure, then $\mathsf{dePol}(\mathbf{X})$ has finite sums and products.*

Proof. (Sketch) We only need to define:

$$\textstyle\sum_I (X_i, Y_i) = (\sum_I X_i, \bigvee_I Y_i) \quad\text{and}\quad \prod_I (X_i, Y_i) = (\bigwedge_I X_i, \prod_I Y_i)$$

as checking these work is trivial. ∎

8.3. Remarks. There are some observations we ought to make about the depolarization construction, and the paper generally.

Depolarization does not depend on the module structure, but is entirely determined by the categories $\mathbf{X}_o$ and $\mathbf{X}_p$. This is probably the reason that the module structure inherent in games goes unremarked upon in most analyses of their structure.

Depolarization is a functor, in a fairly straightforward manner. Any linearly distributive category may be "polarized" in a trivial fashion (*via* the null module on the category). This is left adjoint to depolarization, if we consider these constructions as functors between the category of linearly distributive categories with additive structure and linear functors and linear polarized categories with additive structure and linear polarized functors (as suggested in the conclusion, this is the evident "decorated" version of linear functors). Note that the additive structure is necessary for representability — without it, these results are still true in a poly context.

Given a linear polarized category $\mathbf{X}$ with additive structure, we can construct its game completion, and so we have $\mathsf{dePol}(\mathsf{Gam}_A(\mathbf{X}))$ as well as $\mathsf{dePol}(\mathbf{X})$. Note that $\mathsf{dePol}(\mathbf{X})$ is a full and faithful subcategory of $\mathsf{dePol}(\mathsf{Gam}_A(\mathbf{X}))$.

The tensor in $\mathsf{dePol}(\mathbf{X})$ is essentially that of the combinatorial games of Abramsky and Jagadeesan: since they allow strategies to start with either O or P, they need a pair of maps as in $\mathsf{dePol}(\mathbf{X})$.

We have already remarked that our description in section 1.1 of the AJ combinatorial games was not entirely as they gave them; for one thing, they had a $*$-autonomous category, which we did not, and for another, they did not have explicit polarization in the manner our games have. The construction of the depolarization of a game category brings us closer to their construction, and moreover, restricting to finite strategies, what they construct is $\mathsf{dePol}(\mathsf{Gam}(\mathbb{0}))$, where $\mathbb{0}$ is the null module. We can do a little better, in view of Proposition 8.2.1: we can also include finite cartesian sums and products by freely adding fore additives, to get $\mathsf{dePol}(\mathrm{Fam}(\mathsf{Gam}(\mathbb{0})))$.

In this combinatorial games category, something surprising happens: *viz.* we have units given by $\top = \bigsqcap_\emptyset$ and dually, $\bot = \bigsqcup_\emptyset$. So $\mathbf{1}$ and $\mathbf{0}$ have the rather special properties we noted earlier in discussing the combinatorial games.

We also have some rather unusual "combinatorial games" (*i.e.* polarized game categories) in our setting, since we have not assumed commutativity of the tensor and par. Furthermore, we have a new construction of linearly distributive categories, *viz.* starting with any linearly distributive category with additives, we can polarize and depolarize it, ending up with a new (larger) linearly distributive category.

9. Exponential structure

Following the general principles of this paper, we would expect that exponential structure in the polarized setting will be just a polarized "decoration" of the usual polarized structure. The question, of course, is just how should we type things; the version we shall present will be guided, as has our work so far, by our primary example, the finitary version of Abramsky-Jagadeesan games.

9.1. Exponentials in AJ games. We start by recalling the "Curien-Lamarche exponential" [HS03, C93], which gives exponential structure to (finite) Abramsky-Jagadeesan games. In order to motivate how this structure arises it is useful to recall the so-called "Seely isomorphism":

$$\,!\,(A \wedge B) \equiv \,!\,A \otimes \,!\,B$$

Clearly this should be interpreted in the opponent category. However, notice that, for these combinatorial games, an object in the opponent category may always be written as:

$$Q = \prod_{i\in I} P_i = \bigwedge_{i\in I} P_{i*}$$

where each P_i is a player object and $P_{i*} = \prod_1 P_i$ is the induced opponent object. Then the Seely isomorphism would suggest that the natural first step in the definition of the exponential would be:

$$\,!\,Q = \,!\,\bigwedge_{i\in I} P_{i*} = \bigotimes_{i\in I} \,!\,P_{i*}.$$

This brings us to the issue of how to define $\,!\,P_*$ for an object $P = \bigvee_j Q_j^*$ in the player category. Laurent [L02] quite reasonably used this functor $P \mapsto \,!\,P_*$ as the source of exponentials in his version of polarized games; this is where our treatment of exponentials begins to diverge from his. There are, in fact, different choices which can be made at this point. In the original Abramsky–Jagadeesan paper and subsequently in Hyland–Ong style game semantics one introduces the possibility of infinite games by defining

$$\,!\,P_* = \,!\left(\left(\bigvee_{j\in J} Q_j^*\right)_*\right) = \left(\bigvee_{j\in J} (\,!\,(Q_j \otimes P_*))^*\right)_*.$$

This definition allows the player access to his original state and so the ability to freely backtrack in a computation. This interpretation is neatly accommodated by the Hyland-Ong arenas. It was also the direction that Olivier Laurent chose to follow in his version of polarized games. However, this interpretation takes us outside the finitary AJ games we have been considering, as it requires infinite trees. We could use this if we were to consider a model using infinitary games, but instead we shall follow the Curien-Lamarche approach by making the altogether simpler definition:

$$\,!\,P_* = \,!\left(\left(\bigvee_{j\in J} Q_j^*\right)_*\right) = \left(\bigvee_{j\in J} (\,!\,Q_j)^*\right)_*$$

as this ensures that the exponential is still a finite structure.

Collecting these ideas we define the exponentials for the combinatorial polarized games as follows:

[**!O**]

$$\,!\prod_{j\in J} P_j = \bigotimes_{j\in J} (\,!'P_j)_* \quad \text{where} \quad \,!'\coprod_{i\in I} Q_i = \coprod_{i\in I} \,!\,Q_i$$

[**?P**]

$$\mathbf{?}\bigsqcup_{i\in I} Q_i = \bigoplus_{i\in I}(\,\mathbf{?}'Q_i)^* \quad\text{where}\quad \mathbf{?}'\prod_{j\in J} P_j = \prod_{j\in J}\mathbf{?}\,P_j$$

It is now easy to check the following equalities:

9.1.1. Lemma.

$$\begin{array}{rclrcl}
\mathbf{!}'(Q^*) & = & (\,\mathbf{!}\,Q)^* & \mathbf{?}\,(Q^*) & = & (\,\mathbf{?}'Q)^* \\
\mathbf{!}\,(P_*) & = & (\,\mathbf{!}'P)_* & \mathbf{?}'(P_*) & = & (\,\mathbf{?}\,P)_* \\
\mathbf{!}\,Q_1\otimes\mathbf{!}\,Q_2 & = & \mathbf{!}\,(Q_1\wedge Q_2) & \mathbf{?}\,P_1\oplus\mathbf{?}\,P_2 & = & \mathbf{?}\,(P_1\vee P_2) \\
\mathbf{!}\,\mathbf{1} & = & \mathbf{1} & \mathbf{?}\,\mathbf{0} & = & \mathbf{0}
\end{array}$$

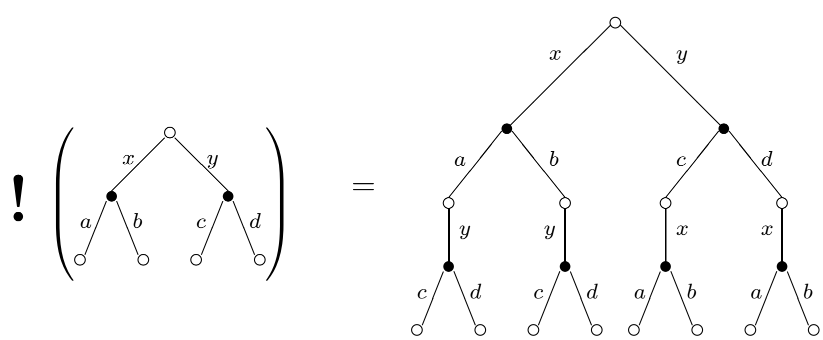
9.1.2. Example. For example, $\mathbf{!}\,(2\colon\{2\colon()\}) = (2\colon\{2\colon(1\colon\{2\colon()\})\})$. Let's work this carefully, to illustrate the definition.

$$\begin{array}{rcl}
\mathbf{!}\,(2\colon\{2\colon()\}) & = & \mathbf{!}\,(\{(),()\},\{(),()\}) \\
 & = & [\,\mathbf{!}'\{(),()\}]_*\otimes[\,\mathbf{!}'\{(),()\}]_* \\
 & = & [\{(),()\}]_*\otimes[\{(),()\}]_* \\
 & = & (\{(),()\})\otimes(\{(),()\}) \\
 & = & (\{(),()\}\otimes(\{(),()\}),(\{(),()\})\otimes\{(),()\}) \\
 & = & (\{(\{(),()\}),(\{(),()\})\},\{(\{(),()\},\{(),()\})\}) \\
 & = & (2\colon\{2\colon(1\colon\{2\colon()\})\})
\end{array}$$

Adding labels, we can see where each move comes from:

$$\begin{array}{rcl}
\mathbf{!}\,(x\colon\{a\colon(),b\colon()\},y\colon\{c\colon(),d\colon()\}) & = & (x\colon\{a\colon(y\colon\{c\colon(),d\colon()\}),b\colon(y\colon\{c\colon(),d\colon()\})\} \\
 & & ,y\colon\{c\colon(x\colon\{a\colon(),b\colon()\}),d\colon(x\colon\{a\colon(),b\colon()\})\})
\end{array}$$

Graphically, this is



where this is precisely the tree of opponent initiated backtrackings.

We can also see the effect of $\epsilon\colon\mathbf{!}\,X\longrightarrow X$ in this case:

$$\begin{array}{rcl}
(x & \mapsto & \overleftarrow{x}\cdot\{a\mapsto\overrightarrow{a}\cdot() \\
 & & \qquad ,b\mapsto\overrightarrow{b}\cdot()\} \\
,y & \mapsto & \overleftarrow{y}\cdot\{c\mapsto\overrightarrow{c}\cdot() \\
 & & \qquad ,d\mapsto\overrightarrow{d}\cdot()\})
\end{array}$$

Table 16: Exponentials

$$\frac{\Gamma, \Gamma' \vdash_x \Delta}{\Gamma, \,!\,Q, \Gamma' \vdash_x \Delta}\, thin \qquad \frac{\Gamma, Q, \Gamma' \vdash_x \Delta}{\Gamma, \,!\,Q, \Gamma' \vdash_x \Delta}\, der$$

$$\frac{!\,\Gamma \vdash_o \,?\,\Delta \,/\, Q \,\backslash\, ?\,\Delta'}{!\,\Gamma \vdash_o \,?\,\Delta \,/\, !\,Q \,\backslash\, ?\,\Delta'}\, stor \qquad \frac{\Gamma, \,!\,Q, \,!\,Q, \Gamma' \vdash_x \Delta}{\Gamma, \,!\,Q, \Gamma' \vdash_x \Delta}\, contr$$

where $\vdash_x$ represents any type of sequent
and the active formula may appear in any position
provided that the typing rules are valid.

Clearly this definition of exponential should come equipped with certain maps (section 9.2): it is standard to require that $!\,(\_)$ be a monoidal comonad for which each object $!\,Q$ is naturally a commutative comonoid as in Seely, Bierman, and ourselves [S87, B95, BCS96]. We shall not present the details of the verification that this "Curien-Lamarche structure" does satisfy all this, (to be honest, some of the calculations are quite tediously involved[!], although the result seems to be folklore), but we shall indicate at least how some of the required structure arises. The comonad structure is more or less straightforward; one uses a structural induction on the formation of the types and decomposes the required maps in a canonical manner. There is a small point: to derive $!\,Q \longrightarrow \,!!\,Q$ one needs, in addition to induction on this, the laxness $!\,O_1 \otimes \,!\,O_2 \xrightarrow{m} \,!\,(O_1 \otimes O_2)$. Establishing this laxness seems to be the most complicated part of the proof; one needs to use duplication $!\,Q \longrightarrow \,!\,Q \otimes \,!\,Q$, and induction. Some key intermediate steps involve establishing $!\,\prod P_i \otimes \,!\,Q \longrightarrow \,!\,\prod(P_i \oslash Q)$, and making heavy use of the decompositions, such as $!\,Q = \,!\,P_{1*} \otimes \cdots \otimes \,!\,P_{n*}$, $Q \otimes Q' = \prod_i (P_i \oslash Q') \wedge \prod_j (Q \otimes P'_j)$, *etc.* With the Seely isomorphism, we get the comonoidal structure maps canonically: $!\,Q \xrightarrow{!\Delta} \,!\,(Q \wedge Q) = \,!\,Q \otimes \,!\,Q$. Note that as the unit $\top$ is the final object $\mathbf{1}$ in this setting, the map $!\,Q \longrightarrow \top$ is forced, as $!\,\mathbf{1} = \mathbf{1}$. The appropriate coherence diagrams are a straightforward (though lengthy) matter to check.

9.2. THE GENERAL SEMANTICS OF EXPONENTIALS. Following the suggestions one may derive from AJ games, we can see that the rules for the exponential (or modal) operators $!$ and $?$ are the evident polarized versions of the rules for these operators in linear logic, $!$ being applied to opponent types and $?$ to player types. There are many ways this principle could be realized; one such is given by the rules in Table 16 (thinning, dereliction, contraction, and storage, using the traditional names) for $!$, with a dual set of rules for $?$. (This presentation has the advantage of indicating what a polarized polycategory with exponentials must be as well.)

Equivalently, in the representable case, this structure may be presented in terms of functors $!$ and $?$ with the following natural transformations and their duals (*i.e.* we have two rules asserting the functoriality of $!$ and $?$, along with axiom schema and their duals as follows). $Q \longleftarrow \,!\,Q \longrightarrow \,!\,!\,Q$ ("$!$ is a cotriple"); $\top \longrightarrow \,!\top$, $!\,Q \otimes \,!\,Q' \longrightarrow \,!\,(Q \otimes Q')$

("$!$ is monoidal"); $!(Q \oplus ?P) \longrightarrow !Q \oplus ?P$ ("$!$ is costrong with respect to $?$"); $\top \longleftarrow !Q \longrightarrow !Q \otimes !Q$ ("free $!$ coalgebras are commutative $\otimes$-comonoids"). The definition of a representable, *i.e.* linear, polarized (poly)category with exponentials must also add the standard coherence conditions, which are suitably "decorated" variants of those found in [BCS96].

We shall not develop this aspect of the theory in complete detail, as it follows the previous pattern (essentially "decorating" the standard structure from linear logic with polarities), instead mainly highlighting where the polarities lie, and what features follow from those decisions. This is especially relevant when we contrast our approach to polarity with that developed by Olivier Laurent.

Let's consider the highlights. We shall say a polarized category $\mathbf{X}$ is equipped with storage if there is a cotriple $!$ on $\mathbf{X}_o$ which is monoidal (with respect to tensor), if the free coalgebras are commutative comonoids, if the dual holds for a triple $?$ on $\mathbf{X}_p$, if $?$ is strong with respect to $!$, and dually $!$ is costrong with respect to $?$, and if the "standard" coherence conditions hold. (These would be "decorated" variants of the conditions spelled out in [BCS96]. The notion of a functor being (co-)strong with respect to a functor of opposite polarity is the typed notion corresponding to the non-polarized notion; since there is only one typing that makes sense, we shall leave the details to the reader.) What is significant about our choices of polarity is that it is possible (and indeed simply using just the standard constructions) to construct the coKleisli category $\mathbf{X}_{o!}$ and the Kleisli category $\mathbf{X}_{p?}$, together with the standard adjunctions; if $\mathbf{X}$ has its module given by an adjunction (*eg.* if $\mathbf{X}$ has unary polarized products and sums at least), then the three adjunctions will fit together, so that we get a polarized category $\mathbf{X}_!$, whose opponent part is $\mathbf{X}_{o!}$, whose player part is $\mathbf{X}_{p?}$, and whose module is given by the composite adjunction, *viz.* arrows of the form $!X \mathrel{-\!\!\!+\!\!\!\rightarrow} ?Y$ in $\widehat{\mathbf{X}}$.

$$\begin{tikzcd}
\mathbf{X}_{o!} \arrow[r, bend right, "!"'] \arrow[r, phantom, "\top"] & \mathbf{X}_o \arrow[l, bend right, "\iota"'] \arrow[r, bend right, "(\ )^*"'] \arrow[r, phantom, "\top"] & \mathbf{X}_p \arrow[l, bend right, "(\ )_*"'] \arrow[r, bend right, "\iota"'] \arrow[r, phantom, "\top"] & \mathbf{X}_{p?} \arrow[l, bend right, "?"']
\end{tikzcd}$$

The monoidal structure in the coKleisli polarized category is cartesian (as is standard in the non-polarized situation). Note also that these definitions and this construction may be carried out in the polarized polycategory setting as well, where the previous remark would then be slightly reworded to say that the polystructure, if representable, is cartesian. This is all pretty straightforward, and it might seem hardly worth mentioning, apart from the observation that the other popular choice of polarities (used in Laurent's work) does not share this feature, and a slightly more roundabout procedure is necessary to capture this coKleisli situation.

Proposition 6.7.2 can be extended to the exponential structure:

9.2.1. Proposition. *Given a polarized polycategory* $\mathbf{X}$ *with a* $!$ *(respectively a* $?$*) then* $\mathsf{Gam}(\mathbf{X})$ *has a* $!$ *(respectively a* $?$*).*

Proof. The lifting of the exponentials follows the Curien-Lamarche definition given in section 9.1.

$$!\prod_{j\in J} P_j = \bigotimes_{j\in J} (!' P_j)_* \quad \text{where} \quad !' \bigsqcup_{i\in I} Q_i = \bigsqcup_{i \in I} !\, Q_i$$

(and dually for $?$). ∎

9.3. Extending exponentials to Fam($\mathbf{X}$). An important extension is to lift our style of exponentials to the family categories, 5.6.

9.3.1. Lemma. *If a polarized game category* $\mathbf{X}$ *has exponentials then* Fam($\mathbf{X}$) *has exponentials.*

Proof. We may define $!\,(X_i)_I = (\,!\,X_i)_I$. This may strike the reader as somewhat unusual, for it suggests (for example) that $!\,(X+Y) = \,!\,X + \,!\,Y$. However, it works. The fact that the product distributes over the added fore coproducts is an important ingredient in making the definition work as can be seen from the following calculation.

$$\begin{aligned} !\,(X \times (Y+Z)) &\cong \;!\,X \otimes \,!\,(Y+Z) \\ &\cong \;!\,X \otimes (\,!\,Y + \,!\,Z) \\ &\cong \;!\,X \otimes \,!\,Y + \,!\,X \otimes \,!\,Z \\ &\cong \;!\,(X \times Y) + \,!\,(X \times Z) \\ &\cong \;!\,(X \times Y + X \times Z) \end{aligned}$$

For example, we may get contraction as follows.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, \,!\,(X_i)_I, \,!\,(X_i)_I, \Gamma' \vdash_x \Delta}{\Gamma, (\,!\,X_i)_I, (\,!\,X_i)_I, \Gamma' \vdash_x \Delta}}{\{\Gamma, \,!\,X_i, \,!\,X_j, \Gamma' \vdash_x \Delta\}_{I \times I}}}{\{\Gamma, \,!\,X_i, \,!\,X_i, \Gamma' \vdash_x \Delta\}_I}}{\{\Gamma, \,!\,X_i, \Gamma' \vdash_x \Delta\}_I}}{\Gamma, (\,!\,X_i)_I, \Gamma' \vdash_x \Delta}}{\Gamma, \,!\,(X_i)_I, \Gamma' \vdash_x \Delta}$$

The other proof rules are handled similarly, and the rules for $?$ are dual. ∎

10. Laurent polarized games

Olivier Laurent [L02] has developed a similar approach to similar material; to conclude this paper we shall compare the logic developed here with that developed under the name "polarized games" and "polarized linear logic" by him. There are several differences, but it is simplest to start with the typing. In Table 17, we list under each polarity the sorts of entities which may be constructed with that polarity. For our games/types, we have continued our use of O for Opponent games or types and P for Player. For Laurent's games/types (which are either Positive or Negative) we have used Pos and Neg (to avoid the clash with P). Note that the best comparison is to be made if one thinks of O as being Pos, and P as being Neg. Another notational "quirk": in the Laurent column we use the standard (Girard-style) linear logic notation, especially for par, but we use the corresponding notation from this paper in our column. The most significant notational difference is that we use $\oplus$ for par, whereas Laurent uses that symbol for coproduct (which we denote by $+$; we have another coproduct, which is denoted $\vee$). The units are also different, as may be seen in Table 17.

Table 17: A tale of two polarizations

	This paper		Laurent	
	O	P	Pos	Neg
Multiplicatives:	$\mathsf{O} \otimes \mathsf{O}$	$\mathsf{P} \oplus \mathsf{P}$	$\mathsf{Pos} \otimes \mathsf{Pos}$	$\mathsf{Neg} \parr \mathsf{Neg}$
Their Units:	$\top$	$\bot$	1	$\bot$
Fore Additives:	$\mathsf{O} + \mathsf{O}$	$\mathsf{P} \times \mathsf{P}$	$\mathsf{Pos} \oplus \mathsf{Pos}$	$\mathsf{Neg} \,\&\, \mathsf{Neg}$
Their Units:	$\sum_\emptyset$	$\prod_\emptyset$	0	$\top$
Aft Additives:	$\mathsf{O} \wedge \mathsf{O}$	$\mathsf{P} \vee \mathsf{P}$	*no*	
Their Units:	1	0		
Polar Additives:	$\mathsf{P} \sqcap \mathsf{P}$	$\mathsf{O} \sqcup \mathsf{O}$	*corresponding*	
Their Units:	1	0		
Mixed tensors:	$\mathsf{O} \mathbin{(\!\oplus} \mathsf{P}$	$\mathsf{P} \mathbin{(\!\otimes} \mathsf{O}$		
	$\mathsf{P} \mathbin{\oplus\!)} \mathsf{O}$	$\mathsf{O} \mathbin{\otimes\!)} \mathsf{P}$	*operations*	
Switch polarity:	$\prod_1 \mathsf{P} = \mathsf{P}_*$	$\coprod_1 \mathsf{O} = \mathsf{O}^*$	$\downarrow \mathsf{Neg}$	$\uparrow \mathsf{Pos}$
	$(\,)^* \dashv (\,)_*$		$\uparrow \;\dashv\; \downarrow$	
Exponentials:	$!\,\mathsf{P}_*$	$?\,\mathsf{O}^*$	$!\,\mathsf{Neg}$	$?\,\mathsf{Pos}$
	$!\,\mathsf{O}$	$?\,\mathsf{P}$	*no corresponding*	
	$?'\,\mathsf{O}$	$!'\,\mathsf{P}$	*operations*	
	$\mathsf{O} \longleftarrow\, !\,\mathsf{O} \longrightarrow\, !\,!\,\mathsf{O}$			
	$\top \longleftarrow\, !\,\mathsf{O} \longrightarrow\, !\,\mathsf{O} \otimes !\,\mathsf{O}$		*no*	
	$\top \longrightarrow\, !\,\top$		*corresponding*	
	$!\,\mathsf{O} \otimes !\,\mathsf{O}' \longrightarrow\, !\,(\mathsf{O} \otimes \mathsf{O}')$		*maps*	
	$!\,(\mathsf{O} \mathbin{(\!\oplus} ?\,\mathsf{P}) \longrightarrow\, !\,\mathsf{O} \mathbin{(\!\oplus} ?\,\mathsf{P}$			
	$!\,\mathsf{O} \otimes !\,\mathsf{O}' \cong\, !\,(\mathsf{O} \wedge \mathsf{O}')$		$!\,\mathsf{Neg} \otimes !\,\mathsf{Neg}' \cong\, !\,(\mathsf{Neg} \,\&\, \mathsf{Neg}')$	

The most evident difference appears almost immediately: we have many more operations. On closer observation, it will also be evident that we have a very different emphasis on the operations. In effect, we make the distinction "conjunction" *vs.* "disjunction", in contrast to Laurent's distinction "multiplicative" *vs.* "additive". (This does seem to be a reflection of the game theoretic origins of the two systems, ours from Abramsky-Jagadeesan games, his from Hyland-Ong games.) Consider the operations we have that Laurent does not have. One pair is not technically significant: as we have pointed out, $\mathsf{P} \sqcap \mathsf{P}$ can be captured from either the aft or the fore additive product and the "inner adjoint" defining the module; in Laurent's style, this would be his $\downarrow \mathsf{Neg} \,\&\, \downarrow \mathsf{Neg}$ (if he had aft products). Since, in his system, as in ours, the fact that $\uparrow \dashv \downarrow$ means that although in general $\mathsf{Pos} \,\&\, \mathsf{Pos}$ is not defined, $\downarrow \mathsf{Neg} \,\&\, \downarrow \mathsf{Neg}$ is "morally" $\cong \downarrow (\mathsf{Neg} \,\&\, \mathsf{Neg})$, and so it may be regarded as "well-formed"; more precisely, we can use $\downarrow (\mathsf{Neg} \,\&\, \mathsf{Neg})$ in its place. In a similar way, $\uparrow \mathsf{Pos} \oplus \uparrow \mathsf{Pos}$ may be defined, and will serve for $\mathsf{Pos} \sqcup \mathsf{Pos}$.

More significant is the missing "mixed" tensor and par operators. In our system they represent the transition between the context and the active formula in a sequent. Although such transitions are present in Laurent's system (note that his sequents have at most one positive formula; that formula, when present, is in effect the "active" formula "in focus", and so is at such a transition), he has no operators representing them. Our point of view is that there is value in explicitly representing all the structure ("all the commas" in effect) at play in one's logic. In addition, there is an immediate benefit: the depolarization construction (section 8) requires these mixed tensors and pars, and (as far as we can determine) there is no corresponding construction for Laurent's system.

Finally, consider the modalities (exponentials) $\boldsymbol{!}$, $\boldsymbol{?}$. The most important feature is that for us, these do not switch polarities (although they are only defined for the appropriate polarities), whereas Laurent's do switch polarity. One consequence of this is that although the "Seely isomorphisms" do hold (in an appropriate sense) in each system, the coKleisli construction, in its most straightforward guise, is only possible in our setup (if $\boldsymbol{!}$ is polarity-switching, one cannot express the cotriple conditions without tinkering with the polarities). Moreover, we have already remarked that the adjunctions "fit together" well, so that the coKleisli construction actually works for polarized categories in our sense, given polarized $\boldsymbol{!}$ and $\boldsymbol{?}$.

Is there a conclusion to be drawn here? Our polarization is based on the product–sum dichotomy, Laurent's on the multiplicative–additive one. Obviously these have some philosophy in common, and some serious philosophical differences. Some constructions work best in one setup, others in the other. We think there may well be a need for both notions; certainly there are models of both.

Most importantly, it is possible to simulate each notion to some degree with the other. In fact there is an adjunction between the two approaches, which we explore next.

10.1. All together now. For the purposes of this section, we shall use the adjectival phrase "Laurent polarized" (and variants) to mean a polarization along the lines of the table above, in particular with fore additives as the primary additive structure, as opposed to our polarization, with its emphasis on our polarized additives. To facilitate comparison we shall identify his Neg with our P, and his Pos with our O, and use our notation throughout (so "par" will be $\oplus$, regardless of which system we are discussing). Our intention is to describe how one may pass between the two notions of polarization, although some of the formalization is being left to the reader.

First we have already observed that any Laurent polarized game category is a polarized game category in our sense, where the polarized sum $\bigsqcup_{i\in I} Q_i$ is defined as $\bigvee_{i\in I} {Q_i}^*$, (in Laurent's notation, this is $\bigoplus_{i\in I} \uparrow Q_i$, *viz.* the coproduct of the images of the Q_is in the "player" category), and the polarized product is defined dually. However, a polarized game category in our sense is not necessarily even a fragment of a Laurent polarized game category, because of the absence of fore additives. Using the polarized version of the "family" construction, we may freely construct $\mathrm{Fam}(\mathbf{X})$ from any polarized game category $\mathbf{X}$ in our sense. We have already seen in sections 5.6, 6.8, and 9.3 how the family construction lifts additive, multiplicative, and exponential structure to a context with fore additives, in other words, to a Laurent polarized context.

A remark here: we lifted our form of exponentials to $\mathrm{Fam}(\mathbf{X})$; but we have already seen that our style of exponentials give rise to the Laurent style, for example, $\mathbf{!}_L P = \mathbf{!}\, P_*$. So we can get Laurent style exponentials in Laurent polarized games, *via* the family construction. It does not seem to be the case, however, that Laurent style exponentials give rise necessarily to our style of exponentials. Note that some of the structure we expect of exponentials does not type in the Laurent setup (as already pointed out above), so they have to be handled very differently in his context.

So, it remains to extend the adjunctions of Propositions 5.6.1, 5.6.2 to the full structure. To summarize these results, we shall first state matters in the poly context; it is clear what it means for a polarized polycategory to have (our style) exponentials (the definition is fairly explicit from Table 16). A polarized functor (or morphism) F of polarized polycategories with exponentials is a polarized exponential functor if it is also a morphism of monads in the following sense: there is a polarized natural transformation, with components $\lambda_o\colon F_o(\mathbf{!}\,Q) \longrightarrow \mathbf{!}\,F_o(Q)$, $\lambda_p\colon \mathbf{?}\,F_p(P) \longrightarrow F_p(\mathbf{?}\,P)$. These must satisfy some obvious coherence conditions, summarized in the representative sample of diagrams below. The reader can supply the usual dual and alternately typed versions.

To motivate these diagrams, let us recall the main structure of the exponentials and of F, which we want λ to preserve (we only list a representative sample of the necessary natural transformations, the rest being generated by duality), which is that $(\mathbf{!}, \mathbf{?})$ is storage and that $(\mathbf{!}, \mathbf{?})$ and (F_o, F_p) are (polarized) linear functors. (The overloaded notation shouldn't cause confusion, given the context of the arrows.)

$\mathbf{!}$ is a cotriple:

$$Q \xleftarrow{\varepsilon} \mathbf{!}\,Q \xrightarrow{\delta} \mathbf{!}\,\mathbf{!}\,Q$$

$\mathbf{!}\,Q$ is a commutative $\otimes$-comonoid:

$$\top \xleftarrow{e} \mathbf{!}\,Q \xrightarrow{d} \mathbf{!}\,Q \otimes \mathbf{!}\,Q$$

$\mathbf{!}$ is monoidal:

$$\top \xrightarrow{m} \mathbf{!}\,\top\ , \quad \mathbf{!}\,Q \otimes \mathbf{!}\,Q' \xrightarrow{m} \mathbf{!}\,(Q \otimes Q')$$

F_o is monoidal:

$$\top \xrightarrow{m} F_o(\top)\ , \quad F_o(Q) \otimes F_o(Q') \xrightarrow{m} F_o(Q \otimes Q')$$

$\mathbf{?}$ is strong with respect to $\mathbf{!}$:

$$\mathbf{?}\,P \mathbin{\langle\!\!\otimes} \mathbf{!}\,Q \xrightarrow{\nu} \mathbf{?}\,(P \mathbin{\langle\!\!\otimes} Q)$$

F_p is strong with respect to F_o:

$$F_p(P) \mathbin{\langle\!\!\otimes} F_o(Q) \xrightarrow{\nu} F_p(P \mathbin{\langle\!\!\otimes} Q)$$

To preserve this structure, we require the following diagrams (and similar diagrams generated by the dualities) to commute.

For the cotriple structure:

$$\begin{tikzcd}
F_o(\mathbf{!}\,Q) \arrow[d, "\lambda"'] \arrow[dr, "F_o(\varepsilon)"] & \\
 & F_o(Q) \\
\mathbf{!}\,F_o(Q) \arrow[ur, "\varepsilon"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
F_o(\mathbf{!}\,Q) \arrow[r, "F_o(\delta)"] \arrow[d, "\lambda"'] & F_o(\mathbf{!!}\,Q) \arrow[d, "\lambda^2"] \\
\mathbf{!}\,F_o(Q) \arrow[r, "\delta"'] & \mathbf{!!}\,F_o(Q)
\end{tikzcd}$$

For the commutative comonoid structure:

$$\begin{tikzcd}
F_o(\mathbf{!}\,Q) \arrow[r, "F_o(e)"] \arrow[d, "\lambda"'] & F_o(\top) \\
\mathbf{!}\,F_o(Q) \arrow[r, "e"'] & \top \arrow[u, "m"']
\end{tikzcd}
\qquad
\begin{tikzcd}
F_o(\mathbf{!}\,Q) \arrow[rr, "F_o(d)"] \arrow[d, "\lambda"'] & & F_o(\mathbf{!}\,Q \otimes \mathbf{!}\,Q) \arrow[r, "F_o(m)"] & F_o(\mathbf{!}\,(Q \otimes Q)) \arrow[d, "\lambda"] \\
\mathbf{!}\,F_o(Q) \arrow[r, "d"] & \mathbf{!}\,F_o(Q) \otimes \mathbf{!}\,F_o(Q) \arrow[r, "m"] & \mathbf{!}\,(F_o(Q) \otimes F_o(Q)) \arrow[r, "\mathbf{!}\,m"] & \mathbf{!}\,F_o(Q \otimes Q)
\end{tikzcd}$$

For the monoidal structure:

$$\begin{tikzcd}
 & F_o(\top) \arrow[r, "F_o(m)"] & F_o(\mathbf{!}\,\top) \arrow[d, "\lambda"] \\
\top \arrow[ur, "m"] \arrow[dr, "m"'] & & \\
 & F_o(\top) \arrow[r, "m"'] & \mathbf{!}\,F_o(\top)
\end{tikzcd}
\qquad
\begin{tikzcd}
 & F_o(\top) \arrow[r, "F_o(m)"] & F_o(\mathbf{!}\,\top) \arrow[d, "\lambda"] \\
\top \arrow[ur, "m"] \arrow[dr, "m"'] & & \\
 & \mathbf{!}\,\top \arrow[r, "\mathbf{!}\,m"'] & \mathbf{!}\,F_o(\top)
\end{tikzcd}$$

$$\begin{tikzcd}
F_o(\,!\,Q) \otimes F_o(\,!\,Q') \arrow[r, "m"] \arrow[d, "\lambda\otimes\lambda"'] & F_o(\,!\,Q \otimes \,!\,Q') \arrow[r, "F_o(m)"] & F_o(\,!\,(Q \otimes Q')) \arrow[d, "\lambda"] \\
!\,F_o(Q) \otimes \,!\,F_o(Q') \arrow[r, "m"'] & !\,(F_o(Q) \otimes F_o(Q')) \arrow[r, "!\,m"'] & !\,F_o(Q \otimes Q')
\end{tikzcd}$$

For the strength:

$$\begin{tikzcd}
 & ?\,F_p(P) \parr \,!\,F_o(Q) \arrow[r, "\nu"] & ?\,(F_p(P) \parr F_o(Q)) \arrow[r, "?\,\nu"] & ?\,F_p(P \parr Q) \arrow[dd, "\lambda"] \\
?\,F_p(P) \parr F_o(\,!\,Q) \arrow[ur, "1\parr\lambda"] \arrow[dr, "\lambda\parr 1"'] & & & \\
 & F_p(\,?\,P) \parr F_o(\,!\,Q) \arrow[r, "\nu"'] & F_p(\,?\,P \parr \,!\,Q) \arrow[r, "F_p(\nu)"'] & F_p(\,?\,(P \parr Q))
\end{tikzcd}$$

10.1.1. Lemma. *If F is a polarized sum–product preserving exponential functor* $\mathbf{X} \longrightarrow \mathbf{Y}$ *between polarized game polycategories with exponentials, the lifting to* $\mathrm{Fam}(F)\colon \mathrm{Fam}(\mathbf{X}) \longrightarrow \mathrm{Fam}(\mathbf{Y})$ *is also a polarized exponential functor.*

Proof. In view of the preceding propositions, the only thing that needs to be proved is that λ lifts to families. This is defined canonically:

$$\begin{array}{rcl}
F(\,!\,(Q_i)_{i\in I}) & \xrightarrow{\text{def}} & F((\,!\,Q_i)_{i\in I}) \\
 & \xrightarrow{\cong} & \sum_{i\in I} F(\,!\,Q_i) \\
 & \xrightarrow{\sum_i \lambda_i} & !\,F(Q_i) \\
 & \xrightarrow{\{!b_i\}_{i\in I}} & !\,\sum_{i\in I} F(Q_i) \\
 & \xrightarrow{\cong} & !\,F((Q_i)_{i\in I})
\end{array}$$

The coherence conditions lift "component-wise" from $\mathbf{X}$. ∎

If we denote the 2-categories consisting of polarized game polycategories, with all additive and exponential structure, together with polarized sum–product preserving exponential functors and polarized polytransformations, by **FPolPolyGam**, and the Laurent-style equivalent by **LFPolPolyGam**, we have the following extension of Proposition 5.6.1.

10.1.2. Proposition. *There is an adjunction*

$$\begin{tikzcd}
\mathbf{FPolPolyGam} \arrow[r, bend right, "\mathsf{Fam}"'] \arrow[r, phantom, "\top"] & \mathbf{LFPolPolyGam} \arrow[l, bend right, "U"']
\end{tikzcd}$$

If we denote the corresponding representable polycategories, with all multiplicative, additive, and exponential structure, by **FPolGam** and **LFPolGam**, then the preceding proposition restricts to the following.

10.1.3. Corollary. *There is an adjunction*

$$\begin{tikzcd} \mathbf{FPolGam} \arrow[r, bend right, "\mathsf{Fam}"'] \arrow[r, phantom, "\top"] & \mathbf{LFPolGam} \arrow[l, bend right, "U"'] \end{tikzcd}$$

11. Concluding remarks

Of course, we hope to find more examples of polarized categories, and to find instances where their structure may help understand other contexts. R. Blute has suggested that an example of our notion of polarization may be found in Borsuk shape theory, for example, and we hope to pursue this with him in the future.

There are several natural extensions of this work, two of which we mention here. First, the technology developed here may be adapted to the non-polarized situation to give a coherent presentation of communication along channels, and in particular, of MALL; this is done for the purely additive fragment in [P03]. Along this direction we could also interpret the polarized structure as polarized processes; the connection is most clearly made *via* a term logic for linear polarized categories (much like that in [P03]). For this to be useful, it is necessary to add other features, most notably fixed points as done by Santocanale.

Also, it is natural to extend the notions developed here to the enriched setting, analogous to Joyal's "money games". We hope to present this in a sequel.

Finally, with the generalization of linearly distributive categories to the polarized setting, the reader familiar with [CS99] might ask if there were a generalization of linear functors to the polarized setting. Indeed there is, with the expected proliferation of components. A linear functor has a tensor and a par component: in the polarized setting, each of these will have an O component, a P component, and a M component. In addition, we shall need polarized variants of the linear strengths, and polarized variants of the coherence conditions given in [CS99]. This notion was not necessary in this paper, the details are routine (if long-winded), and so we shall postpone making all this explicit.

References

[A97] S. Abramsky "Semantics of interaction: an introduction to game semantics", in [PD97] 1–31.

[AJ92] S. Abramsky and R. Jagadeesan "Games and Full Completeness for Multiplicative Linear Logic (Extended Abstract)". Proceedings, Foundations of Software Technology and Theoretical Computer Science (FSTTCS 1992), 291–301.

[AM99] J.-M. Andreoli and R. Maieli "Focussing and proof nets in linear and non-commutative logic". 6th Conference on Logic in Programming and Automated Reasoning (LPAR), Tbilisi, Republic of Georgia, 1999.

[B95] G.M. Bierman "What is a categorical model of intuitionistic linear logic?", in M. Dezani (editor), *Proceedings of Conference on Typed lambda calculus and Applications*, Springer-Verlag, 1995.

[B94] A. Blass "Is Game Semantics Necessary?" Computer Science Logic: 7th Workshop, CSL '93, Springer Lecture Notes in Computer Science **832** (ed. E. Boerger, Y. Gurevich, and K. Meinke), 66–77.

[BCST96] R.F. Blute, J.R.B. Cockett, R.A.G. Seely, and T.H. Trimble "Natural deduction and coherence for weakly distributive categories", Journal of Pure and Applied Algebra **113** (1996) 229–296.

[BCS96] R. Blute, J.R.B. Cockett, and R.A.G. Seely "**!** and **?**: Storage as tensorial strength", Mathematical Structures in Computer Science **6** (1996) 313–351.

[C93] J.R.B. Cockett "Introduction to distributive categories", Mathematical Structures in Computer Science **3** (1993) 277–307.

[C00] J.R.B. Cockett "Game theory revisited: categorical proof theories for games". Talk given at the A.M.S. meeting, Toronto, September 2000. *Notices of the A.M.S.* **47** No.10, November 2000.

[C02a] J.R.B. Cockett "Introduction to polarized games". Tutorial given at C.T.C.S., Ottawa, June 2002.

[C02b] J.R.B. Cockett "Games as polarized categorical proof theory". Talk given at the C.M.S. Summer meeting, Québec, June 2002. `<http://www.cms.math.ca/Events/summer02/abs/>`.

[CKS] J.R.B. Cockett, J. Koslowski, and R.A.G. Seely "Morphisms and modules for poly-bicategories", Theory and Applications of Categories **11** (2003) 15–74.

[CS97a] J.R.B. Cockett and R.A.G. Seely "Weakly distributive categories", Journal of Pure and Applied Algebra **114** (1997) 133–173. (An updated version is available at `<http://www.math.mcgill.ca/rags>`.)

[CS97b] J.R.B. Cockett and R.A.G. Seely "Proof theory for full intuitionistic linear logic, bilinear logic, and MIX categories", Theory and Applications of Categories **3** (1997) 85–131.

[CS99] J.R.B. Cockett and R.A.G. Seely "Linearly distributive functors", Journal of Pure and Applied Algebra **143** (1999) 155–203.

[CS00] J.R.B. Cockett and R.A.G. Seely "Finite sum – product logic", Theory and Applications of Categories **8** (2001) 63–99.

[C93] P.-L. Curien *Categorical combinators, sequential algorithms, and functional programming.* Birkhauser, 2nd ed. 1993.

[G01] J.-Y. Girard "Locus Solum", Mathematical Structures in Computer Science **11** (2001) No. 3, 301–506.

[J95] A. Joyal "Free bicompletion of enriched categories", Math. Reports **XVII**, Acad. Sci. Canada (1995) 213–218.

[HS07] M. Hamano and P.J. Scott "A categorical semantics for polarized MALL", Annals of Pure and Applied Logic **145** (2007) 276–313.

[H00] C. Hermida "Representable multicategories". Advances in Mathematics **151** (2000) 164–225.

[H97] M. Hyland "Game semantics" in [PD97] 131–182.

[HS03] M. Hyland and A. Schalk "Glueing and orthogonality for models of linear logic", Theoretical Computer Science **294** (2003) (1/2) 183–231.

[K82] G.M. Kelly *Basic Concepts of Enriched Category Theory*. London Mathematical Society Lecture Notes Series **64**, Cambridge University Press, 1982.

[L02] O. Laurent "Polarized games", Seventeenth annual IEEE symposium on Logic In Computer Science (LICS '02), 265–274.

[P03] C. Pastro "ΣΠ-polycategories, additive linear logic, and process semantics", MSc Thesis, University of Calgary, submitted 2003.

[PD97] A. Pitts and P. Dybjer, eds. *Semantics and logics of computation*. Publications of the Newton Institute, Cambridge University Press, 1997.

[SS86] S. Schanuel and R. Street "The free adjunction", Cahiers de Topologie et Géométrie Différentielle Catégorique, **27** 81–83.

[S87] R.A.G. Seely "Linear logic, ∗-autonomous categories and cofree coalgebras", in J. Gray and A. Scedrov (editors.), *Categories in Computer Science and Logic*. Contemporary Mathematics **92**, Am. Math. Soc. 1989.

Department of Computer Science, University of Calgary, 2500 University Drive, Calgary, AL, T2N 1N4, Canada
Department of Mathematics and Statistics, McGill University, 805 Sherbrooke St., Montréal, QC, H3A 2K6, Canada

Email: `robin@cpsc.ucalgary.ca` `rags@math.mcgill.ca`