Bi-invariance for Uniform Strategies on Event Structures

Hugo Paquet[1]

Department of Computer Science
University of Oxford
UK

Abstract

A recurring problem in game semantics is to enforce uniformity in strategies. Informally, a strategy is uniform when the Player's behaviour does not depend on the particular indexing of moves chosen by the Opponent. In game semantics, uniformity is used to define a resource modality !, that can be exploited for the semantics of programming languages.

In this paper we give a new account of uniformity for strategies on event structures. This work is inspired by an older idea by Melliès, that uniformity should be expressed as "bi-invariance" with respect to two interacting group actions. We explore the algebraic foundations of bi-invariance, adapt this idea to the language of event structures and define a general notion of uniform strategy in this context. Finally we revisit the existing approach to uniformity, and show how this arises as a special case of our constructions.

Keywords: Game semantics, linear logic, event structures, uniformity, group actions, distributive laws

1 Introduction

This paper is about the foundations of game semantics, and in particular the notion of uniformity for strategies. Informally, uniformity is the property that a strategy does not depend on the particular indexing of moves by the Opponent. This is a key component in various kinds of game semantics [1,5,13], necessary to obtain a cartesian closed model and thus validate basic soundness properties.

In this paper we explore an algebraic description of uniformity, inspired by an unpublished manuscript by Melliès ([11], 2003). Melliès argues for a treatment of uniformity as "bi-invariance" under two interacting group actions. We take this idea further:

- we give an abstract definition of bi-invariance in categorical terms;
- we propose a new, general notion of uniform strategy based on event structures; and
- we formally relate this to the state-of-the-art approach to uniformity in concurrent games [7], shedding new light on existing definitions of uniformity.

In §1.1 we give an informal introduction to uniformity, using some examples of uniform and non-uniform strategies for a simple concrete game. Then we introduce event structures, games and strategies (§1.2), and we formally motivate uniformity and bi-invariance (§1.3).

---

[1] Email: hugo.paquet@cs.ox.ac.uk

## 1.1   Uniform and non-uniform strategies

Consider a two-player game where the only possible action for each player is to put tokens down on the table. Each player has access to an infinite supply of tokens; tokens can be put down at any time and in any order; and the tokens are indistinguishable. So the game consists of

$$\bullet_0 \quad \bullet_1 \quad \bullet_2 \quad \cdots \qquad\qquad \bullet_0 \quad \bullet_1 \quad \bullet_2 \quad \cdots \tag{1}$$

where Blue and Red are the two players and their respective tokens are indexed by $\mathbb{N}$. We want to reason about strategies for Blue (regardless of winning conditions for the game). Here are three possible strategies:

(i) "Put a token down every time Red puts a token down."

(ii) "Never put down any tokens."

(iii) "Put a token down without waiting for Red, and *two more* every time Red puts down a token."

And here is a possible graphical representation for each of them:
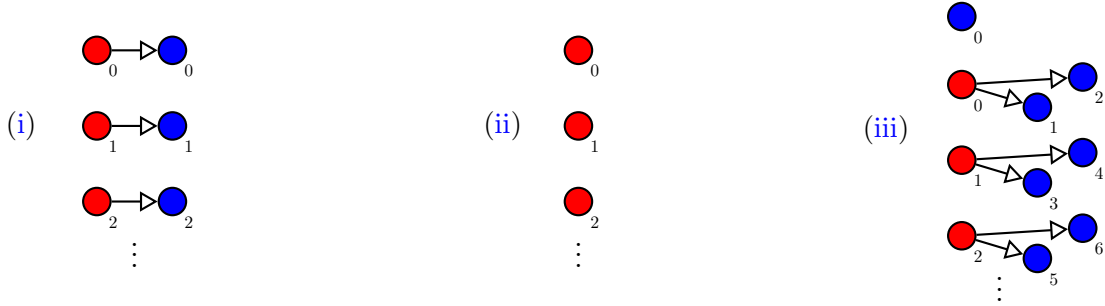
Arrows specify the dependency of Blue actions on Red actions. The absence of an arrow between two moves means that they may happen in any order or at the same time. This is true in particular for Red moves: Blue does not know which specific tokens Red will choose to play.

For Blue, the choice of which token to put down at a given point is arbitrary: tokens are indistinguishable. Strategy (i) does not explicitly require that Blue plays the same token number as Red, and Blue's choice does not affect the course of the game.

But if the choice of token does not matter, then our representation should not allow for the following strategies:

(iv) "Put a token down only if Red puts down token number 2."

(v) "Put n tokens down after Red plays token number n."

We say that these strategies are **non-uniform**, because the behaviour of Blue is dependent on the token that Red chooses to play. In more general games, uniformity is difficult to express and reason about.

1.2   Concurrent games

We are interested in a particular mathematical theory of games and strategies, based on event structures, initially developed by Rideau and Winskel [16]. These games are often called "concurrent games" because

of the possibility for players to perform multiple actions at the same time, as in strategy (iii) above.

Concurrent games can be presented in a clear way as a double category (e.g. [10]), with components:

- *objects*: event structures.
- *vertical morphisms* $A \to B$: maps of event structures.
- *horizontal morphisms* $A \nrightarrow B$: strategies from A to B.
- *2-cells*: maps of strategies.

We define each component in turn.

Event structures.

An event structure is a partial order $(A, \leq)$ in which each $a \in A$ has a finite number of predecessors, equipped with a polarity function $A \to \{-, +\}$, and an irreflexive and symmetric conflict relation $\#$ which is hereditary: if $a \leq a'$ and $a \# b$ then $a' \# b$. The operational intuition is that elements of A are the possible moves in a game, with the polarity function assigning each move to one of the two players. Moves can only be played when their predecessors in $\leq$ have been played, and moves related by $\#$ cannot occur in the same play.

Therefore the possible states of the game A are the finite subsets $x \subseteq A$ which are down-closed and contain no two moves in conflict; these are called **configurations**. The set of configurations is written $\mathscr{C}(A)$. For example, the game in (1) is an event structure where both $\leq$ and $\#$ are trivial. (Note: we simply call *event structure* what is typically called an *event structure with polarity*.)

Maps of event structures.

A map of event structures $(A, \leq, \#) \longrightarrow (B, \leq, \#)$ is a function $f : A \to B$ such that for every $x \in \mathscr{C}(A)$, $fx \in \mathscr{C}(B)$ and moreover f is injective when restricted to x. The idea is that any execution of A can be faithfully simulated in B. Maps also have to preserve polarity.

Strategies.

A strategy from A to B consists of an event structure σ together with a map of event structures

$$\sigma \xrightarrow{\ p_\sigma\ } A^\perp \parallel B$$

where the operation $-^\perp$ flips the polarity of every event, and $\parallel$ is a monoidal product representing the parallel composition of event structures. The idea is that every execution allowed by the strategy σ must correspond to a play of the game. Strategies must satisfy a technical condition which we omit for now and explain below (Definition 3.1).

An important special case is when A is the empty game. Then we say that σ is a strategy on the game B. For example, all the strategies described in §1.1 are strategies on the game described in (1), where the arrows $\twoheadrightarrow$ correspond to $\leq$.

Maps of strategies.

Let σ and τ be strategies from A to B and from C to D, respectively. Suppose that $\alpha : \sigma \to \tau$ makes the following diagram commute, for maps of event structures f and g:

$$
\begin{array}{ccc}
\sigma & \xrightarrow{\ \alpha\ } & \tau \\
{\scriptstyle p_\sigma}\big\downarrow & & \big\downarrow{\scriptstyle p_\tau} \\
A^\perp \parallel B & \xrightarrow[\ f^\perp \parallel g\ ]{} & C^\perp \parallel D
\end{array}
$$

Then we say α is a map of strategies from σ to τ, with boundary f and g. These are the 2-cells of the double category. This double category is weak (or *pseudo*), because the composition of strategies is not strictly associative or unital, only up to invertible 2-cells.

Remark 1.1 The specific details of double categories are not important for this paper. Most of this work is only concerned with strategies over a fixed game A, their internal symmetries, and notions of maps between strategies over A. We only mention the categorical framework in this introduction because it explains the need for uniformity, as we describe next.

1.3   Duplication of moves and uniformity

The monoidal structure given by parallel composition $\|$ is not cartesian: this is a linear model. This is not surprising because game semantics is closely tied to linear logic. In many applications, for example to give semantics to a higher-order programming language, we resolve this by constructing a resource modality, typically a (pseudo-)comonad !. Briefly, $!A$ is a duplicated version of A in which every move is available in countably many copies. For example, the game in (1) is of the form $!A$, where A is a game with a single move for each player.

But when defining ! naively we run into issues of uniformity. The difficulty is to properly account for the symmetries that arise in the manipulation of copies, as we explain now.

For an event structure A, the object $!A$ is an infinitary parallel composition:

$$!A = \coprod_{i \in \omega} A.$$

From any bijection $\alpha : \omega \to \omega^2$, it is possible to construct a family of "co-multiplication" strategies $\delta_A : !A \nrightarrow !!A$. Unfortunately, for fixed α the associativity law

$$
\begin{array}{ccc}
!A & \xrightarrow{\ \delta_A\ } & !!A \\
{\scriptstyle \delta_A}\downarrow & & \downarrow{\scriptstyle \delta_{!A}} \\
!!A & \xrightarrow[\ !\delta_A\ ]{} & !!!A
\end{array}
\tag{2}
$$

does not hold. We need a way to indicate that the different copies of A in $!A$ (and $!!A$, *etc.*) are exchangeable. This does not make the diagram commute strictly, but it does up to a notion of permutation of moves.

Permuting moves gives an equivalence relation on strategies. For soundness, this relation must be stable under composition. This requires uniformity: for example, in the non-uniform strategy (iv) in §1.1, the result of the game is not the same when the opponent Red plays token 1 or token 2.

On the other hand, consider the uniform strategy (i). If Red decides to update from token 1 to token 2, then the result is the same, although Blue must also permute tokens, to follow the strategy. This is what is called **bi-invariance** in the work of Melliès [11]: for every Opponent permutation, there is a *response* Player permutation under which the strategy is invariant.

1.4   Context and related work

Uniformity in concurrent games.
    Castellan, Clairambault and Winskel have shown how to address the uniformity problem in concurrent games by upgrading to *event structures with symmetry*, a generalisation introduced by Winskel [20] precisely to handle constructions such as $!A$. Adding symmetry to concurrent games is challenging and there are several approaches ([5,6]), but a consensus has been reached around a framework known as *thin concurrent games* [7]. Applications of thin concurrent games are already far-reaching [8,15].

While successful, this construction is technical. Symmetry is often described as a "proof-relevant equivalence relation", which indicates which moves are copies of each other via a so-called isomorphism family, *i.e.* a family of bijections between configurations. This means that the construction of the model must account for these families of bijections, both in games and strategies, and this requires additional conditions relating them.

We will see how uniform strategies on thin concurrent games arise as a special case of our more general notion of uniform strategies. In particular, we move away from event structures with symmetry and isomorphism families, although we show how they can be recovered (§6).

Uniformity in game semantics in general.
The need for uniformity arises in any game model which makes duplication explicit. For instance, the pioneering game semantics of Abramsky, Jagadeesan, and Malacaria [1] enforce uniformity via a partial equivalence relation on strategies. This method is too restrictive for the concurrent strategies we consider.

Many models are based on *plays with pointers*, following Hyland and Ong [9]. The uniformity problem does not arise there, because with pointers one can avoid explicit duplication. One drawback is that the underlying linear structure is harder to see. Melliès explains this phenomenon in [11], motivating his work on asynchronous games; another explanation is given by Castellan and Clairambault in the preprint [4], which also connects with thin concurrent games.

There is a well-known intersection of ideas between asynchronous games and thin concurrent games, but an important difference is in the treatment of uniformity. These two lines of research are brought closer together as a by-product of this paper.

Outline of the paper.
In §2 we study symmetry in games in terms of two compatible groups actions. This is based on existing ideas [11,7], but the presentation in terms of distributive laws is new to this paper. In §3, we explain how to reindex moves in strategies using so-called weak maps between them. Then, in §4 we introduce uniform strategies, after developing some algebraic principles for bi-invariance. In §5 we focus on the copycat strategy, a cornerstone of game semantics, and show that it is uniform. Finally, in §6, we make a formal connection between the contributions of this paper and the established theory of thin concurrent games.

2   Games and permutations

Our first step is to consider games (*i.e.* event structures) equipped with algebraic structure encoding the symmetries. Our presentation uses the definition of a group as a set G equipped with maps

$$1 \xrightarrow{e} G \qquad\qquad G \times G \xrightarrow{m} G \qquad\qquad G \xrightarrow{inv} G$$

satisfying unit, associativity, and inverse laws. We will also use a basic fact about groups:

Lemma 2.1 *For any group G, the functor $G \times (-) : \mathbf{Set} \to \mathbf{Set}$ has a canonical monad structure, induced by the maps m and e. An algebra over this monad is a set with a left action of G.*

More concretely, a left action of G on a set A is a map $act : G \times A \to A$ such that the diagrams

$$
\begin{array}{ccc}
1 \times A & & \\
{\scriptstyle e \times A}\big\downarrow \ \ \searrow{\scriptstyle \cong} & & \\
G \times A \xrightarrow[act]{} A & &
\end{array}
\qquad\qquad
\begin{array}{ccc}
(G \times G) \times A & \xrightarrow{\ m \times A\ } & G \times A \\
{\scriptstyle \cong}\big\downarrow & & \big\downarrow{\scriptstyle act} \\
G \times (G \times A) \xrightarrow[G \times act]{} G \times A \xrightarrow[act]{} A
\end{array}
$$

commute. Every group element $g \in G$ induces an automorphism $act(g, -)$ of the set A. Indeed an action is equivalently defined as a group homomorphism $G \to \mathrm{Aut}(A)$, where $\mathrm{Aut}(A)$ is the automorphism group

of A. This definition makes sense for A an object of any category. The action is called *faithful* if distinct elements induce distinct automorphisms, and in this case the group G can be identified with a subgroup of $\mathrm{Aut}(A)$.

2.1 Positive and negative automorphisms

We will consider group actions on games. If A is a game, an automorphism of event structures $\theta : A \to A$ describes a way to permute moves while preserving the dependency and conflict structure.

Two configurations of A are considered symmetric when the action of group element substitutes one with the other. A game can have different symmetry structures, and so we can interpret the same game in different ways. For example, there are exactly two automorphisms of the game A below: the identity map, and the map swapping the left and right columns.

$$\tag{3}$$

If the swapping bijection is allowed, then this game is understood as consisting of two exchangeable copies of the same game. Otherwise, A is just a game of the form $B \parallel B$.

In this paper, the symmetry structure on a game A will consist of two groups $\mathcal{N}_A$ and $\mathcal{P}_A$, both acting on A on the left. The idea is that elements of $\mathcal{N}_A$ represent permutations of negative (Red, Opponent) moves, and elements of $\mathcal{P}_A$ are permutations of positive (Blue, Player) moves. Formally the situation is more subtle: since automorphisms of A must preserve the causal structure, in general negative automorphisms have a non-trivial action on positive moves, and vice-versa.

For configurations x, y of an event structure A, we write $x \subseteq^+ y$ if $x \subseteq y$ and all moves in $y \setminus x$ are positive; $\subseteq^-$ is defined similarly.

Definition 2.2 Let A be an event structure with polarity. An automorphism $\alpha \in \mathrm{Aut}(A)$ is **negative** if it satisfies the following condition: for every $x \in \mathcal{C}(A)$, if α fixes x, and $x \subseteq^+ y$, then α fixes y. Similarly, α is **positive** if, whenever α fixes x, and $x \subseteq^- y$, then α fixes y.

An action of a group G on A is **negative** (resp. **positive**) if every group element induces a negative (resp. positive) automorphism.

Example 2.3 The swapping bijection for the game in (3) is negative, but not positive: it fixes the empty configuration $\emptyset$, but extensions $\emptyset \subseteq^- x$ are not fixed. Note that the Player moves are swapped too, but intuitively this is forced by Opponent.

Next we define games. The purpose of the definition is to axiomatize the interaction between positive and negative symmetries in the game.

2.2 Games equipped with group actions

Definition 2.4 A **game** is an event structure A equipped with:

- a group $\mathcal{N}$ and a negative group action $act_\mathcal{N} : \mathcal{N} \times A \to A$,
- a group $\mathcal{P}$ and a positive group action $act_\mathcal{P} : \mathcal{P} \times A \to A$,
- and a distributive law $\lambda : \mathcal{N} \times \mathcal{P} \to \mathcal{P} \times \mathcal{N}$ between the monads $\mathcal{N} \times (-)$ and $\mathcal{P} \times (-)$ on **Set**,

such that the actions are permuted by λ, as in the diagram below:

$$
\begin{array}{ccc}
(\mathcal{N} \times \mathcal{P}) \times A & \xrightarrow{\;\lambda \times A\;} & (\mathcal{P} \times \mathcal{N}) \times A \\
\cong \downarrow & & \downarrow \cong \\
\mathcal{N} \times (\mathcal{P} \times A) & & \mathcal{P} \times (\mathcal{N} \times A) \\
\mathcal{N} \times act_{\mathcal{P}} \downarrow & & \downarrow \mathcal{P} \times act_{\mathcal{N}} \\
\mathcal{N} \times A & \xrightarrow[act_{\mathcal{N}}]{} A \xleftarrow[act_{\mathcal{P}}]{} & \mathcal{P} \times A.
\end{array}
$$

Distributive laws between groups

We make some comments about this definition. A distributive law between monads S and T is usually defined as a natural transformation $\lambda_X : ST(X) \to TS(X)$, compatible with the monad structures as specified by four axioms [3].

A natural transformation $\mathcal{N} \times (\mathcal{P} \times X) \to \mathcal{P} \times (\mathcal{N} \times X)$ is determined by its component at $X = 1$, and so it can be presented as a combinator $\lambda : \mathcal{N} \times \mathcal{P} \to \mathcal{P} \times \mathcal{N}$, as we have done in the definition. The axioms for a distributive law can then be given directly in terms of the group structure in $\mathcal{N}$ and $\mathcal{P}$:

$$
\begin{array}{ccccccc}
1 \times \mathcal{P} & \xrightarrow{\;\cong\;} & \mathcal{P} \times 1 & \qquad & \mathcal{N} \times 1 & \xrightarrow{\;\cong\;} & 1 \times \mathcal{N} \\
e \times \mathcal{P} \downarrow & & \downarrow \mathcal{P} \times e & & \mathcal{N} \times e \downarrow & & \downarrow e \times \mathcal{N} \\
\mathcal{N} \times \mathcal{P} & \xrightarrow[\lambda]{} & \mathcal{P} \times \mathcal{N} & & \mathcal{N} \times \mathcal{P} & \xrightarrow[\lambda]{} & \mathcal{P} \times \mathcal{N}
\end{array}
$$

$$
\begin{array}{ccc}
 & \mathcal{N} \times (\mathcal{P} \times \mathcal{N}) & \\
\xrightarrow{\mathcal{N} \times \lambda} & & \searrow^{\cong} \\
\mathcal{N} \times (\mathcal{N} \times \mathcal{P}) & & (\mathcal{N} \times \mathcal{P}) \times \mathcal{N} \\
\cong \downarrow & & \downarrow \lambda \times \mathcal{N} \\
(\mathcal{N} \times \mathcal{N}) \times \mathcal{P} & & (\mathcal{P} \times \mathcal{N}) \times \mathcal{N} \\
 & & \downarrow \cong \\
m \times \mathcal{P} \downarrow & & \mathcal{P} \times (\mathcal{N} \times \mathcal{N}) \\
 & & \downarrow \mathcal{P} \times m \\
\mathcal{N} \times \mathcal{P} & \xrightarrow{\qquad\qquad \lambda \qquad\qquad} & \mathcal{P} \times \mathcal{N}
\end{array}
\qquad
\begin{array}{ccc}
 & (\mathcal{P} \times \mathcal{N}) \times \mathcal{P} & \\
\xrightarrow{\lambda \times \mathcal{P}} & & \searrow^{\cong} \\
(\mathcal{N} \times \mathcal{P}) \times \mathcal{P} & & \mathcal{P} \times (\mathcal{N} \times \mathcal{P}) \\
\cong \downarrow & & \downarrow \mathcal{P} \times \lambda \\
\mathcal{N} \times (\mathcal{P} \times \mathcal{P}) & & \mathcal{P} \times (\mathcal{P} \times \mathcal{N}) \\
 & & \downarrow \cong \\
\mathcal{N} \times m \downarrow & & (\mathcal{P} \times \mathcal{P}) \times \mathcal{N} \\
 & & \downarrow m \times \mathcal{N} \\
\mathcal{N} \times \mathcal{P} & \xrightarrow{\qquad\qquad \lambda \qquad\qquad} & \mathcal{P} \times \mathcal{N}
\end{array}
$$

A combinator λ of this kind is also known as a *Zappa-Szép product* of the groups $\mathcal{N}$ and $\mathcal{P}$ [19].

Permuting subgroups and factorization.

When $\mathcal{N}$ and $\mathcal{P}$ are subgroups of the same group, for example if the actions on A are faithful and $\mathcal{N}, \mathcal{P}$ are seen as subgroups of $\mathrm{Aut}(A)$, then the combinator λ is uniquely determined provided $\mathcal{N}$ and $\mathcal{P}$ are *permuting subgroups*:

$$
\text{for every } \alpha \in \mathcal{N} \text{ and } \beta \in \mathcal{P}, \text{ there exist } \alpha' \in \mathcal{N} \text{ and } \beta' \in \mathcal{P} \text{ such that } \alpha\beta = \beta'\alpha'. \tag{4}
$$

In this case, α' and β' are unique since $\mathcal{N}$ and $\mathcal{P}$ have trivial intersection, and so we must have $\lambda(\alpha, \beta) = (\beta', \alpha')$. All games of interest in semantics seem to have this property, maybe because in practice many group actions are faithful.

The property (4) arises in thin concurrent games as a factorization property [7]. There is a tight relationship between distributive laws, Zappa-Szép products, and strict factorization systems, which explains this [17]. It is useful in this paper to have an explicit combinator λ, as part of the structure of a game.

Any event structure can be seen as a game with trivial symmetry. In this case none of the moves in A are exchangeable. More interesting games arise from the constructions we describe next.

Dual games.

Any game A has a dual game $A^\perp$. The underlying event structure is the same, with polarity reversed. The symmetry structure on $A^\perp$ is defined by noticing that for any game the combinator $\lambda : \mathcal{N} \times \mathcal{P} \to \mathcal{P} \times \mathcal{N}$ gives rise to a combinator $\mathcal{P} \times \mathcal{N} \longrightarrow \mathcal{N} \times \mathcal{P}$ in a canonical way. We will call $isw : \mathcal{P} \times \mathcal{N} \to \mathcal{N} \times \mathcal{P}$ the invert-and-swap mapping $(\alpha, \beta) \mapsto (inv(\beta), inv(\alpha))$.

Lemma 2.5 *For a game A, the event structure $A^\perp$ equipped with $\mathcal{N}_{A^\perp} = \mathcal{P}_A$, $\mathcal{P}_{A^\perp} = \mathcal{N}_A$, and the combinator*

$$\lambda_{A^\perp} = \mathcal{P}_A \times \mathcal{N}_A \xrightarrow{isw} \mathcal{N}_A \times \mathcal{P}_A \xrightarrow{\lambda} \mathcal{P}_A \times \mathcal{N}_A \xrightarrow{isw} \mathcal{N}_A \times \mathcal{P}_A,$$

*is a game known as the **dual** of A.*

Parallel composition: a tensor product of games.

Two games A and B can be combined using the parallel composition of event structures:

Definition 2.6 *For event structures A and B, the **parallel composition** $A \parallel B$ is the event structure with events $A \uplus B$, and with $\leq$, $\#$, and pol inherited from A and B.*

Since there is no conflict between events of A and B, every configuration of $A \parallel B$ is of the form $x_A \parallel x_B$, and so there is a canonical isomorphism $\mathscr{C}(A \parallel B) \cong \mathscr{C}(A) \times \mathscr{C}(B)$. We use this to define the symmetry structure in $A \parallel B$.

Lemma 2.7 *Define the **parallel composition** of games A and B as the event structure $A \parallel B$, equipped with groups $\mathcal{N}_{A\parallel B} = \mathcal{N}_A \times \mathcal{N}_B$ and $\mathcal{P}_{A\parallel B} = \mathcal{P}_A \times \mathcal{P}_B$ having the product action on $\mathscr{C}(A \parallel B) \cong \mathscr{C}(A) \times \mathscr{C}(B)$. The combinator $\lambda_{A\parallel B}$ is given by*

$$\mathcal{N}_{A\parallel B} \times \mathcal{P}_{A\parallel B} \xrightarrow{\cong} (\mathcal{N}_A \times \mathcal{P}_A) \times (\mathcal{N}_B \times \mathcal{P}_B) \xrightarrow{\lambda_A \times \lambda_B} (\mathcal{P}_A \times \mathcal{N}_A) \times (\mathcal{P}_B \times \mathcal{N}_B) \xrightarrow{\cong} \mathcal{P}_{A\parallel B} \times \mathcal{N}_{A\parallel B}.$$

This satisfies the axioms for a game.

A resource modality.

In game semantics, types are interpreted as games whose initial moves all have the same polarity, and depending on this polarity we call games of this kind either *negative* or *positive*. The idea is that, for a negative game A, the copies of A in $!A$ can be permuted only by elements of $\mathcal{N}_{!A}$.

Lemma 2.8 *For a negative game A, the event structure $!A = \parallel_{i\in\omega} A$ is equipped with groups*

$$\mathcal{N}_{!A} = \{(\pi, (\alpha_i)_{i\in\omega}) \mid \pi \text{ is a permutation of } \omega, \text{ and } \alpha_i \in \mathcal{N}_A \text{ for all } i \in \omega\}$$
$$\mathcal{P}_{!A} = \{(\alpha_i)_{i\in\omega} \mid \alpha_i \in \mathcal{P}_A \text{ for all } i \in \omega\}$$

under componentwise multiplication, with action on $!A$ defined by

$$act_\mathcal{N}((\pi, (\alpha_i)_{i\in\omega}), (i, a)) = (\pi(i), \alpha_i(a))$$
$$act_\mathcal{P}((\alpha_i)_{i\in\omega}, (i, a)) = (i, \alpha_i(a))$$

and distributive law

$$\lambda_{!A} : \mathcal{N}_{!A} \times \mathcal{P}_{!A} \longrightarrow \mathcal{P}_{!A} \times \mathcal{N}_{!A}$$
$$((\pi, (\alpha_i)_{i\in\omega}), (\beta_i)_{i\in\omega}) \longmapsto \left(\left(\beta'_{\pi^{-1}(i)} \right)_{i\in\omega}, (\pi, (\alpha'_i)_{i\in\omega}) \right)$$

where for every $i \in \omega$, $(\beta'_i, \alpha'_i) = \lambda_A(\alpha_i, \beta_i)$. This satisfies the axioms for a game.

Other important constructions are involved in applications of game semantics, including a cartesian product $A \,\&\, B$ and a linear function space $A \multimap B$ (e.g. [7]). We omit them here, because the focus is on algebraic foundations rather than applications, and these constructions do not interfere with the group actions.

Global and local reindexing maps

We discuss a point of notation. The group actions associated to a game A induce automorphisms of the game. For $\alpha \in \mathcal{N}$, we will write $\alpha : A \to A$ for the induced automorphism. This is an abuse of notation but should cause no confusion. Such a map α is a *global* reindexing of the moves of A.

We often also need to discuss the *local* reindexing of a specific configuration, and so for $x \in \mathcal{C}(A)$ we will write $\alpha : x \to y$ to mean the restriction of α to x, where $y = \alpha x$ is the image of x. We will do this for any map of event structures. Note that the local restriction is always a bijection, because maps of event structures must be locally injective.

3   Strategies and permutations

We consider strategies on a fixed game A. The definition is due to Rideau and Winskel [16]:

Definition 3.1 A **strategy** on a game A consists in an event structure σ, together with a projection map $p_\sigma : \sigma \to A$ such that, for every $x \in \mathcal{C}(\sigma)$,

- if $p_\sigma x \subseteq^- z$, there is a unique $y \in \mathcal{C}(\sigma)$ such that $x \subseteq y$ and $p_\sigma y = z$, and
- if $z \subseteq^+ p_\sigma x$, there is a (necessarily unique) $y \in \mathcal{C}(\sigma)$ such that $y \subseteq x$ and $p_\sigma y = z$.

We write $\sigma : A$ to mean that σ is a strategy on A.

These axioms do not play a major role in this paper: informally they say that a strategy (for Player) should not restrict the behaviour of Opponent or the order in which Opponent processes the Player moves. Next we define a basic notion of maps between strategies [16]:

Definition 3.2 A **strict map of strategies** from $\sigma : A$ to $\tau : A$ is a map of event structures $f : \sigma \longrightarrow \tau$ which commutes with the projection maps:

$$
\begin{array}{ccc}
\sigma & \xrightarrow{\ f\ } & \tau \\
& {\scriptstyle p_\sigma}\searrow \quad \swarrow {\scriptstyle p_\tau} & \\
& A &
\end{array}
$$

We will now give a more general notion of map. With a strict map as above, we have that $p_\sigma x = p_\tau f x$ for every configuration $x \in \mathcal{C}(\sigma)$. Since our games now have symmetry, we can relax this equality and allow for a local reindexing of Player moves via a positive bijection $f[x]$.

Definition 3.3 A **weak map of strategies** from $\sigma : A$ to $\tau : A$ consists of a map $f : \sigma \longrightarrow \tau$ together with, for every $x \in \mathcal{C}(\sigma)$, a positive automorphism $f[x] \in \mathcal{P}$ such that the following diagram commutes:

$$
\begin{array}{ccc}
x & \xrightarrow{\ f\ } & fx \\
{\scriptstyle p_\sigma}\downarrow & & \downarrow{\scriptstyle p_\tau} \\
p_\sigma x & \xleftarrow[\ f[x]\ ]{} & p_\tau f x.
\end{array}
$$

This gives a generalized notion of maps, with strict maps as a special case. Importantly, two strategies are now isomorphic when the underlying event structures are isomorphic and the projections agree up to a Player permutation. (This would allow us to define an associativity 2-cell filling the diagram (2).)

Example 3.4 The simplest examples of non-strict weak maps of strategies are those $f : \sigma \to \tau$ for which $f[x] = f[y]$ for every $x, y \in \mathscr{C}(\sigma)$. This means that the Player reindexing is *global*, and we then have $p_\sigma = \beta \circ p_\tau \circ f$ for some $\beta \in \mathcal{P}$. (Conversely, if the latter is true, then f is a weak map of strategies with every $f[x] = \beta$.)

We write $\mathrm{Strat}(A)$ for the category of strategies on A and weak maps between them, with identities and composition defined in terms of those in $\mathcal{P}$. The next lemma is an easy observation that will be important for the next section.

Lemma 3.5 *There is a functor* $\mathrm{Strat}(A) \longrightarrow \mathbf{Set}_{\mathcal{P} \times (-)}$, *where* $\mathbf{Set}_{\mathcal{P} \times (-)}$ *is the Kleisli category for the monad* $\mathcal{P} \times (-)$, *that sends a strategy* σ *to the set* $\mathscr{C}(\sigma)$ *and a weak map* $f : \sigma \to \tau$ *to the Kleisli function*

$$\mathscr{C}(\sigma) \longrightarrow \mathcal{P} \times \mathscr{C}(\tau)$$
$$x \longmapsto (f[x], fx).$$

This functor is faithful, although not injective on objects.

4   Uniform strategies

We introduce our notion of uniformity. The approach is guided by the suggestion ([11]) that uniformity can be understood as a group-theoretic property. Informally, a uniform strategy should be invariant under any permutation of Opponent moves, *up to* a permutation of Player moves. We show how to express this formally.

4.1   Permutations of Opponent moves

Let $\sigma : A$ be a strategy. For every $\alpha \in \mathcal{N}$, we can build a new strategy, that has the same underlying event structure σ but with projection map

$$\sigma \xrightarrow{p_\sigma} A \xrightarrow{\alpha} A.$$

This is a version of σ in which moves have been permuted according to α. This operation forms a left action of $\mathcal{N}$ on the set of strategies on A. This new strategy (the result of α acting on σ) is denoted $\alpha \cdot \sigma$.

The goal of uniformity is to ensure that this action leaves the strategy unchanged, except for a relabelling of Player moves as permitted by $\mathcal{P}$. One challenge is that the Player response to some Opponent permutation $\alpha \in \mathcal{N}$ may not be consistent across the strategy, and so we need to reason locally.

To enforce uniformity, we will ask for a weak map of strategies $\phi_\alpha : \alpha \cdot \sigma \to \sigma$, for every $\alpha \in \mathcal{N}$. Each ϕ_α determines a map

$$\mathscr{C}(\sigma) \longrightarrow \mathcal{P} \times \mathscr{C}(\sigma) \tag{5}$$

which records the image of every configuration and a suitable local response to α. (As a consequence, the action of $\mathcal{N}$ on strategies extends to a functorial operation on weak maps. This is not true for weak maps between non-uniform strategies.)

4.2   Algebras for bi-invariance

We discuss an algebraic formalization of bi-invariance with respect to two groups and a distributive law between them. For this section, assume $\mathcal{N}, \mathcal{P}$ and λ are fixed and arbitrary.

Recall (§2) that sets with a left action of the group $\mathcal{N}$ are precisely algebras for the monad $\mathcal{N} \times (-)$ on **Set**. To account for the Player response in (5), the key idea is to move to the Kleisli category $\mathbf{Set}_{\mathcal{P} \times (-)}$.

We recall the following basic fact about distributive laws ([3]):

Lemma 4.1 *If $\mathcal{P}$ and $\mathcal{N}$ are groups with a distributive law $\lambda : \mathcal{N} \times \mathcal{P} \to \mathcal{P} \times \mathcal{N}$, the monad $\mathcal{N} \times (-)$ on* **Set** *lifts to a monad on the Kleisli category* **Set**$_{\mathcal{P} \times (-)}$.

An algebra over the lifted monad consists of a set X together with a map $\mathcal{N} \times X \to \mathcal{P} \times X$. This is what we take as a basic model for the bi-invariance of X with respect to $\mathcal{N}$ and $\mathcal{P}$.

The functor part of this monad takes a set X to $\mathcal{N} \times X$, and a Kleisli map $X \xrightarrow{f} \mathcal{P} \times Y$ to the map $\mathcal{N} \times X \to \mathcal{P} \times (\mathcal{N} \times Y)$ obtained by post-composing $\mathcal{N} \times f$ with $\lambda \times Y$, modulo associativity isomorphisms. Rather than explaining the full monad structure, we give the axioms for its algebras:

Lemma 4.2 *A pair $(X, \mathcal{N} \times X \xrightarrow{h} \mathcal{P} \times X)$ is an algebra over the monad $\mathcal{N} \times (-)$ on* **Set**$_{\mathcal{P} \times (-)}$ *whenever the following equations hold in* **Set***, writing $(\mu^{\mathcal{P}}, \eta^{\mathcal{P}})$ and $(\mu^{\mathcal{N}}, \eta^{\mathcal{N}})$ for the canonical monad structures:*

$$
\begin{array}{ccc}
\mathcal{N} \times (\mathcal{N} \times X) \xrightarrow{\mathcal{N} \times h} \mathcal{N} \times (\mathcal{P} \times X) \xrightarrow{\lambda_X} \mathcal{P} \times (\mathcal{N} \times X) \\
\downarrow{\scriptstyle \mathcal{P} \times h} \\
\mu^{\mathcal{N}}_X \downarrow \qquad\qquad\qquad\qquad\qquad \mathcal{P} \times (\mathcal{P} \times X) \\
\downarrow{\scriptstyle \mu^{\mathcal{P}}_X} \\
\mathcal{N} \times X \xrightarrow{\qquad\qquad h \qquad\qquad} \mathcal{P} \times X
\end{array}
\qquad
\begin{array}{ccc}
& X & \\
\eta^{\mathcal{N}}_X \downarrow & & \searrow{\scriptstyle \eta^{\mathcal{P}}_X} \\
\mathcal{N} \times X & \xrightarrow{\quad h \quad} & \mathcal{P} \times X
\end{array}
$$

Next we use this algebraic structure to define uniform strategies.

4.3 Uniform strategies

Definition 4.3 A **uniform strategy** on a game A is a strategy σ equipped with a weak map $\phi_\alpha : \alpha{\cdot}\sigma \to \sigma$ for every $\alpha \in \mathcal{N}$, such that the induced map

$$
\phi : \mathcal{N} \times \mathscr{C}(\sigma) \longrightarrow \mathcal{P} \times \mathscr{C}(\sigma)
$$

defined by $\phi(\alpha, x) = (\phi_\alpha[x], \phi_\alpha x)$ makes $(\mathscr{C}(\sigma), \phi)$ an algebra over the monad $\mathcal{N} \times (-)$ on **Set**$_{\mathcal{P} \times (-)}$.

Since ϕ determines the family of weak maps $(\phi_\alpha)_{\alpha \in \mathcal{N}}$, we denote a uniform strategy by a pair (σ, ϕ). It is helpful to unfold the definition. For any $\alpha \in \mathcal{N}$ and $x \in \mathscr{C}(\sigma)$, we have a diagram

$$
\begin{array}{ccc}
x & \xrightarrow{\quad \phi_\alpha \quad} & y \\
\downarrow & & \downarrow \\
p_\sigma x & \xrightarrow{\ \alpha\ } \alpha p_\sigma x \xleftarrow{\ \phi_\alpha[x]\ } & p_\sigma y
\end{array}
$$

which must commute because ϕ_α is a weak map. To unfold the composition axiom, consider $\alpha, \alpha' \in \mathcal{N}$. We apply the distributive law, to get $(\gamma, \beta) = \lambda(\alpha', \phi_\alpha[x])$, and the overall situation is depicted as

where the top component of the diagram must commute, and $\gamma \circ \phi_\beta[y] = \phi_{\alpha'\alpha}[x]$, so that the bottom-right triangle commutes.

We now ensure that weak maps are compatible with the uniformity structure.

Definition 4.4 A **weak map of uniform strategies** from (σ, ϕ) to (τ, ψ) is a weak map $f : \sigma \to \tau$ such that the induced map $\mathscr{C}(\sigma) \to \mathcal{P} \times \mathscr{C}(\tau)$ is an algebra homomorphism $(\mathscr{C}(\sigma), \phi) \to (\mathscr{C}(\tau), \psi)$.

Algebra homomorphisms are closed under composition, and so there is a category $\mathrm{UStrat}(A)$ of uniform strategies over a game A, and a functor into the category of algebras. Summarizing, we have

$$
\begin{array}{ccc}
\mathrm{UStrat}(A) & \xrightarrow{\;\mathscr{C}(-)\;} & \mathbf{Set}^{\mathcal{N}\times(-)}_{\mathcal{P}\times(-)} \\
\downarrow & & \downarrow \\
\mathrm{Strat}(A) & \xrightarrow[(cf.\ 3.5)]{\mathscr{C}(-)} & \mathbf{Set}_{\mathcal{P}\times(-)}
\end{array}
$$

where the vertical arrows are forgetful functors.

5 The copycat strategy: definition and applications

The copycat strategy is the identity morphism on a game A. It is a strategy $A \to A$, so a strategy on the game $A^\perp \parallel A$. The idea is to play a positive move on one side as soon as the corresponding negative move has been played on the other [16]:

Definition 5.1 The event structure $\mathbb{c}_A$ has the same events as $A^\perp \parallel A$, and the same polarity and conflict relation, but $\leq_{\mathbb{c}_A}$ is the transitive closure of the relation $\leq_{A^\perp\parallel A} \cup \{((0,a),(1,a)) \mid \mathrm{pol}_A(a) = +\} \cup \{((1,a),(0,a)) \mid \mathrm{pol}_A(a) = -\}$.

The **copycat strategy** on A is the event structure $\mathbb{c}_A$ equipped with the map $p_{\mathbb{c}_A} : \mathbb{c}_A \to A^\perp \parallel A$ with identity action on events.

The $\mathbb{c}$ construction extends to a functorial operation: for every map of event structures $f : A \to B$, the map $f^\perp \parallel f : A^\perp \parallel A \to B^\perp \parallel B$ can also be seen as a map $\mathbb{c}_f : \mathbb{c}_A \to \mathbb{c}_B$.

5.1 The uniform copycat strategy

We equip the copycat strategy on A with a map

$$
\phi : \mathcal{N}_{A^\perp\parallel A} \times \mathscr{C}(\mathbb{c}_A) \longrightarrow \mathcal{P}_{A^\perp\parallel A} \times \mathscr{C}(\mathbb{c}_A)
$$

as required for uniformity. The construction of ϕ is straightforward using the axioms for a game. Let $(\beta, \alpha) \in \mathcal{N}_{A^\perp\parallel A} = \mathcal{P}_A \times \mathcal{N}_A$. The key observation is that β and α can be made to agree using the distributive law: if $(\gamma^{-1}, \delta) = \lambda(\alpha, \beta^{-1})$, then (regarding group elements as automorphisms) the diagram

$$
\begin{array}{ccc}
 & A \xrightarrow{\;\beta\;} A \xrightarrow{\;\delta\;} & \\
A & & A \\
 & \xrightarrow{\alpha} A \xrightarrow{\gamma} &
\end{array}
$$

commutes. We can set $\varepsilon := \delta \circ \beta = \gamma \circ \alpha$, so that the diagram below commutes:

$$
\begin{array}{ccc}
\mathbb{c}_A & \xrightarrow{\;\;\mathbb{c}_\varepsilon\;\;} & \mathbb{c}_A \\
p\downarrow & & \downarrow p \\
A^\perp \parallel A \xrightarrow[\beta\parallel\alpha]{} A^\perp \parallel A & \xrightarrow[\delta\parallel\gamma]{} & A^\perp \parallel A
\end{array}
$$

By the discussion in Example 3.4, cc_ε is a weak map $(\beta, \alpha) \cdot \mathrm{cc}_A \to \mathrm{cc}_A$. We set $\phi_{(\beta,\alpha)} := \mathrm{cc}_\varepsilon$.

Lemma 5.2 *The pair (cc_A, ϕ), as defined in the preceding discussion, defines a uniform strategy.*

5.2 Lifting maps of event structures to strategies

Copycat has other applications beyond providing identity morphisms. For certain well-behaved maps of event structures $f : A \to B$, we can use copycat to lift f to a *strategy* $\hat{f} : A \nrightarrow B$ ([23,14,22]). This is often used to lift basic structural morphisms such as the symmetry of parallel composition $A \parallel B \to B \parallel A$, or a co-multiplication $!A \to !!A$.

Definition 5.3 Let $f : A \to B$ be a map of event structures which satisfies the conditions for A to be a strategy on B, with projection map f. Then the strategy $\hat{f} : A \nrightarrow B$ is defined as the event structure cc_A with projection map

$$\mathrm{cc}_A \xrightarrow{p_{\mathrm{cc}_A}} A^\perp \parallel A \xrightarrow{A^\perp \parallel f} A^\perp \parallel B.$$

The purpose of this section is to identify sufficient conditions on f under which $\hat{f}$ can be made uniform. The basic requirement is that f should reflect the negative group action and preserve the positive group action, in a functorial and coherent way.

Lemma 5.4 *Let $f : A \to B$ be a map of event structures as in Definition 5.3, such that there are group homomorphisms $L : \mathcal{N}_B \to \mathcal{N}_A$ and $M : \mathcal{P}_A \to \mathcal{P}_B$ such that for every $\alpha \in \mathcal{N}_B$ and $\beta \in \mathcal{P}_A$,*

$$
\begin{array}{ccc}
A & \xrightarrow{L(\alpha)} & A \\
f\downarrow & & \downarrow f \\
B & \xrightarrow{\alpha} & B
\end{array}
\qquad\qquad
\begin{array}{ccc}
A & \xrightarrow{\beta} & A \\
f\downarrow & & \downarrow f \\
B & \xrightarrow{M(\beta)} & B
\end{array}
$$

and additionally the following coherence property holds:

$$
\begin{array}{c}
\mathcal{N}_B \times \mathcal{P}_A \xrightarrow{\mathcal{N}_B \times M} \mathcal{N}_B \times \mathcal{P}_B \xrightarrow{\lambda_B} \mathcal{P}_B \times \mathcal{N}_B \xrightarrow{\mathcal{P}_B \times L} \mathcal{P}_B \times \mathcal{N}_A \\
\mathcal{N}_B \times \mathcal{P}_A \xrightarrow{L \times \mathcal{P}_A} \mathcal{N}_A \times \mathcal{P}_A \xrightarrow{\lambda_A} \mathcal{P}_A \times \mathcal{N}_A \xrightarrow{M \times \mathcal{N}_A} \mathcal{P}_B \times \mathcal{N}_A
\end{array}
$$

Then, the strategy $\hat{f}$ can be made uniform with $\phi_{\hat{f}}$ defined as the composite

$$\mathcal{N}_{A^\perp \parallel B} \times \mathscr{C}(\mathrm{cc}_A) \xrightarrow{(\mathcal{N}_{A^\perp} \times L) \times \mathscr{C}(\mathrm{cc}_A)} \mathcal{N}_{A^\perp \parallel A} \times \mathscr{C}(\mathrm{cc}_A) \xrightarrow{\phi_{\mathrm{cc}_A}} \mathcal{P}_{A^\perp \parallel A} \times \mathscr{C}(\mathrm{cc}_A) \xrightarrow{(\mathcal{P}_{A^\perp} \times M) \times \mathscr{C}(\mathrm{cc}_A)} \mathcal{P}_{A^\perp \parallel B} \times \mathscr{C}(\mathrm{cc}_A).$$

Remark 5.5 The lifting construction has a *co-lifting* counterpart: in some cases, for a map $f : A \to B$, the composite

$$\mathrm{cc}_A \xrightarrow{p_{\mathrm{cc}_A}} A^\perp \parallel A \xrightarrow{f^\perp \parallel A} B^\perp \parallel A.$$

gives a strategy from B to A, which can be made uniform providing f is compatible with symmetry in a way that is essentially dual to Lemma 5.4.

A key application of these results is in the definition of a comonad structure for $!$. For example the counit $!A \nrightarrow A$ is obtained by co-lifting the injection $A \to !A$. Liftings and co-liftings of this kind have a universal property [18] (see also [14]), which provides canonical constructions for 2-cells between lifted maps, as required for (2). We do not develop this further, as this paper focuses on the study of uniformity.

6   Thin concurrent games

We explore the connection between the algebraic development in this paper and the established theory of thin concurrent games. We recall the main definitions of games, strategies, and maps of strategies from [7,6], and explain how they arise within our setting.

6.1   Symmetry in event structures and games

First we focus on the games. Permutation of moves in thin concurrent games is described using isomorphism families [20]:

Definition 6.1 Let E be an event structure. An **isomorphism family** on E is a set $\mathcal{S}(E)$ of polarity-preserving bijections between configurations of E, containing identities and closed under composition and taking inverses, such that for every $(\theta : x \cong y) \in \mathcal{S}(E)$:

(restriction) if $x' \subseteq x$, then the restriction of θ to x' is in $\mathcal{S}(E)$; and

(extension) if $x \subseteq x'$, then there exists some $\theta' : x' \cong y'$ in $\mathcal{S}(E)$ such that θ is the restriction of θ' to x.

The pair $(E, \mathcal{S}(E))$ is then called an **event structure with symmetry**. Maps of event structures with symmetry are also required to preserve symmetry.

Castellan, Clairambault and Winskel define a notion of game with symmetry [7]:

Definition 6.2 A **thin concurrent game** is an event structure A equipped with three isomorphism families $\mathcal{S}(A), \mathcal{S}_+(A)$, and $\mathcal{S}_-(A)$ such that $\mathcal{S}_+(A), \mathcal{S}_-(A) \subseteq \mathcal{S}(A)$, and:

- if $\theta \in \mathcal{S}_+(A) \cap \mathcal{S}_-(A)$, then $\theta = \mathrm{id}_x$ for some $x \in \mathscr{C}(A)$;
- if $\theta \in \mathcal{S}_+(A)$ and $\theta \subseteq^+ \theta' \in \mathcal{S}(A)$, then $\theta' \in \mathcal{S}_+(A)$; and
- if $\theta \in \mathcal{S}_-(A)$ and $\theta \subseteq^- \theta' \in \mathcal{S}(A)$, then $\theta' \in \mathcal{S}_-(A)$.

With the next result we begin to explain the connection with the work in this paper.

Lemma 6.3   (i) *Let G be a group acting on an event structure E. Then the set*

$$\mathcal{S}_G(E) = \{\theta : x \cong y \mid \theta \text{ is the restriction to } x \text{ of some } E \xrightarrow{g} E, g \in G\}$$

defines an isomorphism family on E.

(ii) *Let A be a game in the sense of Definition 2.4. Then the structure of A defines a thin concurrent game $\underline{A}$, where $\mathcal{S}_-(\underline{A}) = \mathcal{S}_\mathcal{N}(A)$, $\mathcal{S}_+(\underline{A}) = \mathcal{S}_\mathcal{P}(A)$, and $\mathcal{S}(\underline{A})$ is the closure of $\mathcal{S}_-(\underline{A}) \cup \mathcal{S}_+(\underline{A})$ under composition.*

We have shown that our games give rise to thin concurrent games. It seems that all useful thin concurrent games arise in this way, although there are pathological examples, which arise because isomorphism families provide bisimulations rather than automorphisms.

6.2   Symmetry in strategies

In thin concurrent games, strategies are also equipped with isomorphism families.

Definition 6.4 Let A be a thin concurrent game. A $\sim$**-strategy** on A is an event structure with symmetry σ, together with a map $(\sigma, \mathcal{S}(\sigma)) \to (A, \mathcal{S}(A))$ whose underlying map is a strategy, satisfying a further two axioms:

(thinness) If $x \in \mathscr{C}(\sigma)$, and $\mathrm{id}_x \subseteq^+ \theta \in \mathcal{S}(\sigma)$, then $\theta = \mathrm{id}_{x'}$ for some $x' \in \mathscr{C}(\sigma)$.

($\sim$-receptivity) If $x \subseteq^- y, z \in \mathscr{C}(\sigma)$, and there exists some $(\theta : py \cong pz) \in \mathcal{S}(A)$ such that $\mathrm{id}_{px} \subseteq \theta$, then there exists $\chi \in \mathcal{S}(\sigma)$ such that $\mathrm{id}_x \subseteq \chi$ and $p\chi = \theta$.

As we will see below (Proposition 6.7), we can explain these axioms by imposing a *locality* requirement on the uniformity. To achieve this, we connect our uniform strategies to $\sim$-strategies. In the next lemma, we show that for every uniform strategy $\sigma : A$, it is possible to recover an isomorphism family on σ.
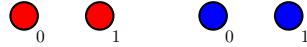
Lemma 6.5 *Let (σ, ϕ) be a uniform strategy on a game A. Then*

$$\mathcal{S}_\phi(\sigma) = \{\theta : x \cong y \mid \theta \text{ is the restriction to } x \text{ of } \phi_\alpha, \text{ for some } \alpha \in \mathcal{N}\}$$

is an isomorphism family on the event structure σ, and the map $p_\sigma : \sigma \to A$ is a map of event structures with symmetry $(\sigma, \mathcal{S}_\phi(\sigma)) \to (A, \mathcal{S}(\underline{A}))$.

Unfortunately, thinness and $\sim$-receptivity do not hold in general, as the next example shows.

Example 6.6 Consider the simple game A given by the event structure

where $\mathcal{N} = \{\alpha, \mathrm{id}\}$ and $\mathcal{P} = \{\beta, \mathrm{id}\}$ are the symmetry groups for the negative and positive events, respectively, with the canonical action, and trivial distributive law.

Consider the strategy $\sigma = A$ where p_σ is the identity map. We construct a bi-invariance structure ϕ. By the axioms, ϕ_{id} must be the identity map on σ. Define $\phi_\alpha = \beta \circ \alpha$, so that swapping the Opponent moves also triggers a swap of the Player moves. This satisfies the axioms for a uniform strategy, since both α and $\beta \circ \alpha$ are their own inverse.

But the family $\mathcal{S}_\phi(\sigma)$, obtained by restricting ϕ_α and ϕ_{id} to individual configurations, contains a bijection

 $\qquad\qquad\qquad\qquad$ (6)

which breaks thinness, as a non-identity positive extension of the identity on $\emptyset$. (It is also not $\sim$-receptive, taking x to contain the positive moves, and y, z the two one-event negative extensions of x.)

We give an informal analysis of the problem: by providing only a *local* account of symmetry, the isomorphism family $\mathcal{S}_\phi(\sigma)$ fails to record that the bijection (6) comes from ϕ_α, and forgets that it is only valid when the Opponent moves are swapped too. In this local view, it looks as though the two positive moves are swapped for no reason. This is resolved by the next result.

Proposition 6.7 *Let (σ, ϕ) be a uniform strategy on A such that ϕ satisfies the following property:*

(uniformity is local.) *For all $x \in \mathcal{C}(\sigma)$ and $\alpha \in \mathcal{N}$, if α fixes $p_\sigma x$, then ϕ_α fixes x.*

Then, $(\sigma, \mathcal{S}_\phi(\sigma))$ is a $\sim$-strategy on $\underline{A}$, written $\underline{\sigma}_\phi$.

In particular, uniformity in Example 6.6 is not local. This way, we have identified the strategies on thin concurrent games with a restricted, well-behaved class of uniform strategies, in which all uniformity information is accessible locally. We note that it is *not* the case that every uniform strategy can be assigned a modified ϕ for which uniformity is local. Thus we have a proper generalization.

6.3   Weak maps of strategies

Definition 6.8 Let A be a thin concurrent game and let σ, τ be $\sim$-strategies. A **weak map of $\sim$-strategies** [7] is a map $f : \sigma \to \tau$ of event structures with symmetry, such that for every $x \in \mathcal{C}(\sigma)$ there exists a (necessarily unique) bijection $(\theta : p_\sigma x \cong p_\tau f x) \in \mathcal{S}_+(A)$ such that the following diagram commutes:

$$\begin{array}{ccc} x & \xrightarrow{\ f\ } & fx \\ {\scriptstyle p_\tau}\downarrow & & \downarrow{\scriptstyle p_\tau} \\ p_\sigma x & \xrightarrow[\ \theta\ ]{} & p_\tau f x \end{array} \qquad\qquad (7)$$

As expected, we obtain this by restricting the weak maps between uniform strategies.

Lemma 6.9 *Let (σ, ϕ) and (τ, ψ) be uniform strategies on a game A, and let $f : \sigma \to \tau$ be a map of uniform strategies between them. Then:*

(i) *f is a map of event structures with symmetry $(\sigma, \mathcal{S}_\phi(\sigma)) \to (\tau, \mathcal{S}_\psi(\tau))$; and*

(ii) *for every $x \in \mathscr{C}(\sigma)$ there exists $\theta \in \mathcal{S}_P(A)$ such that the diagram in (7) commutes.*

As an immediate corollary, we obtain that if uniformity is local in (σ, ϕ) and (τ, ψ), a weak map f of uniform strategies induces a weak map $\underline{f} : \underline{\sigma}_\phi \to \underline{\tau}_\psi$ between $\sim$-strategies on the thin concurrent game $\underline{A}$.

7   Concluding remarks

We have defined a new general notion of uniform strategies on event structures. Our presentation has made explicit the algebraic structures and coherence laws underlying the usual treatment of uniformity in game semantics. The work is specific to concurrent games, but the algebraic perspective could facilitate the transfer of ideas to other kinds of games, e.g. [12].

Within the landscape of concurrent games, this development will combine smoothly with weights or annotations on event structures, as required to model other kinds of programming (e.g. [21,2]): it suffices to require that each ϕ_α preserves the weights.

This work should be seen as a preliminary investigation. It remains to determine how far we can take the uniform strategies in this paper, in full generality. In particular, adapting the current methods for composing strategies will require some work, as these rely strongly on the local perspective [7]. This is an important challenge, because the main purpose of uniformity (*cf.* §1.3) is to ensure that permutations of copies interact well with the composition of strategies.

Acknowledgement

References

[1] Samson Abramsky, Radha Jagadeesan, and Pasquale Malacaria. Full abstraction for PCF. *Information and Computation*, 163(2):409–470, 2000.
https://doi.org/10.1006/inco.2000.2930.

[2] Aurore Alcolei. *Jeux concurrents enrichis: témoins pour les preuves et les ressources.* PhD thesis, ENS Lyon, 2019.
https://perso.ens-lyon.fr/aurore.alcolei/these.pdf

[3] Jon Beck. Distributive laws. In *Seminar on triples and categorical homology theory*, pages 119–140. Springer, 1969.
https://doi.org/10.1007/BFb0083084

[4] Simon Castellan and Pierre Clairambault. Disentangling parallelism and interference in game semantics. *CoRR* (2021)
https://doi.org/10.48550/arxiv.2103.15453.

[5] Simon Castellan, Pierre Clairambault, and Glynn Winskel. Symmetry in concurrent games. In *Proceedings of the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–10, 2014.
https://doi.org/10.1145/2603088.2603141

[6] Simon Castellan, Pierre Clairambault, and Glynn Winskel. The parallel intensionally fully abstract games model of PCF. In *2015 30th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 232–243. IEEE, 2015.
https://doi.org/10.1109/LICS.2015.31

[7] Simon Castellan, Pierre Clairambault, and Glynn Winskel. Thin games with symmetry and concurrent Hyland-Ong games. *Logical Methods in Computer Science*, 2019.
https://doi.org/10.48550/arXiv.1409.7542

[8] Pierre Clairambault and Marc de Visme. Full abstraction for the quantum lambda-calculus. *Proceedings of the ACM on Programming Languages*, 4(POPL):1–28, 2019.
https://doi.org/10.1145/3371131

[9] J Martin E Hyland and C-HL Ong. On full abstraction for PCF: I, II, and III. *Information and computation*, 163(2), 2000.
https://doi.org/10.1006/inco.2000.2917

[10] Niles Johnson and Donald Yau. *2-dimensional categories*. Oxford University Press, 2021.
https://doi.org/10.1093/oso/9780198871378.001.0001

[11] Paul-André Melliès. Asynchronous games 1: Uniformity by group invariance, 2003, preprint

[12] Paul-André Melliès. Asynchronous template games and the gray tensor product of 2-categories. In *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*, pages 1–13. IEEE, 2021.
https://doi.org/10.1109/LICS52264.2021.9470758

[13] Paul-André Melliès and Samuel Mimram. Asynchronous games: Innocence without alternation. In *International Conference on Concurrency Theory*, pages 395–411. Springer, 2007.
https://doi.org/10.1007/978-3-540-74407-8_27

[14] Hugo Paquet. *Probabilistic concurrent game semantics*. PhD thesis, University of Cambridge, 2020.
https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-950.pdf

[15] Hugo Paquet and Glynn Winskel. Continuous probability distributions in concurrent games. *Electronic Notes in Theoretical Computer Science*, 341:321–344, 2018.
https://doi.org/10.1016/j.entcs.2018.11.016

[16] Silvain Rideau and Glynn Winskel. Concurrent strategies. In *2011 IEEE 26th Annual Symposium on Logic in Computer Science*, pages 409–418. IEEE, 2011.
https://doi.org/10.1109/LICS.2011.13

[17] Robert Rosebrugh and Richard J Wood. Distributive laws and factorization. *Journal of Pure and Applied Algebra*, 175(1-3):327–353, 2002.
https://doi.org/10.1016/S0022-4049(02)00140-8

[18] Michael Shulman. Framed bicategories and monoidal fibrations. *Theory and Applications of Categories*, 20(18):650–738, 2008.
http://www.tac.mta.ca/tac/volumes/20/18/20-18.pdf

[19] Jenö Szép. On the structure of groups which can be represented as the product of two subgroups. *Acta Sci. Math. Szeged*, 12(Leopoldo Fejer et Frederico Riesz LXX annos natis dedicatus, Pars A):57–61, 1950.
http://acta.bibl.u-szeged.hu/13587/1/math_012_pars_a_057-061.pdf

[20] Glynn Winskel. Event structures with symmetry. *Electronic Notes in Theoretical Computer Science*, 172:611–652, 2007.
https://doi.org/10.1016/j.entcs.2007.02.022

[21] Glynn Winskel. Distributed probabilistic and quantum strategies. *Electr. Notes Theor. Comput. Sci.*, 298:403–425, 2013.
https://doi.org/10.1016/j.entcs.2013.09.024

[22] Glynn Winskel. Making concurrency functional. *CoRR*, abs/2202.13910, 2022.
https://doi.org/10.48550/arXiv.2202.13910

[23] Glynn Winskel, Silvain Rideau, Pierre Clairambault, and Simon Castellan. Games and strategies as event structures. *Logical Methods in Computer Science*, 13, 2017.
https://doi.org/10.23638/LMCS-13(3:35)2017