Non-angelic concurrent game semantics

Simon Castellan[1], Pierre Clairambault[1], Jonathan Hayman[2], and Glynn Winskel[2]

[1]Univ Lyon, CNRS, ENS de Lyon, UCB Lyon 1, LIP, Lyon, France
[2]Computer Laboratory, University of Cambridge, Cambridge, U.K.

***Abstract*—The *hiding operation*, crucial in the construction of categories of games and strategies and hence the compositional aspect of game semantics, has a tendency, as a side effect, to remove branches of computation not leading to observable results. Accordingly, games models of programming languages are usually biased towards *angelic* non-determinism, where branches leading to *e.g.* divergence are forgotten.**

We present here new categories of games, which do not suffer from this bias. In our first category, we achieve this by avoiding hiding altogether; instead morphisms are *uncovered* strategies (with neutral/invisible events) up to *weak bisimulation*. Then, we show that by hiding only certain events dubbed *inessential* we can consider strategies up to *isomorphism*, and still get a category – this partial hiding remains sound up to weak bisimulation, so we get a concrete representations of morphisms (as in standard concurrent games) while avoiding the angelic bias.

We give a semantics for Affine Idealized Parallel Algol which is adequate for both may and must equivalence within the model.

I. Introduction

A longstanding issue when giving semantics to nondeterministic processes is at what level of abstraction should divergence, the process entering an internal loop, be captured: possible operational choices are to record each individual internal step, to record simply that the process has the potential to diverge in a given state, or even to completely disregard the possibility. In this paper, we study a range of choices in the setting of concurrent games and the effect that these have when they are used as the basis of game semantics.

By modelling the possible ways in which processes interact with their context, game semantics makes it possible to obtain compositional semantics for languages including features such as higher-order processes and concurrency. For nondeterministic languages, most effort has been put into representations which are *angelic*, where the representation of interaction disregards the possibility of divergence and only records when processes *may* converge. This loss of a handle on the possibility of divergence means that such semantics are not adequate for *must-convergence*. For instance the term M = `if choice tt` $\bot$ that makes a nondeterministic choice between converging and diverging and the term N = `tt` that always converges have the same interpretation. However, N must converge whereas M might not.

In this paper, we use concurrent games based on event structures to develop the non-angelic game semantics of a simple prototypical concurrent, higher-order, shared-memory language. In comparison to a standard interleaved trace-set semantics, event structure-based semantics has two convenient features when studying the possibility of divergence: they explicitly record when processes may branch and, by not reducing concurrency to sets of possible interleavings, it is not necessary to describe when an interleaved trace is fair.

Contributions of the paper: The starting representation for our semantics is to allow the representation of processes to include internal 'hidden' events. Such events can be used to record how processes may diverge: the occurrence of an internal event might inhibit the occurrence of all the non-hidden events that would otherwise indicate progress. The category of concurrent games introduced in [14] can be extended to retain hidden events in the composition of strategies, giving this model a very operational flavour. However, the category obtained is only a compact-closed category when strategies are viewed up to weak bisimulation. As a consequence of viewing strategies up to weak bisimulation, an interpretation of a language in this model amounts to a giving an operational semantics by means of a labelled transition system with an in-built notion of independence of events. The semantics ensures that weak bisimulation is a congruence with respect to interaction.

The construction of the interpretation of terms gives rise to progressively large event structures as the terms grow, containing internal events that are redundant from the perspective of keeping track of divergence. We specify which internal events are *essential* to obtain a more compact representation: composition now hides all inessential internal events. In doing so, we get back a category up to isomorphism without losing behaviours up to weak bisimulation.

Related work: Harmer [10] uses *stopping traces* to record where strategies can get stuck, providing a game semantics capturing both may and must convergence. Towards achieving the benefits described above of an event structure semantics, a similar methodology is adopted in [7] by replacing stopping traces by stopping configurations of event structures. However, this approach is tailored to must-equivalence and it is not clear how it would scale to other testing equivalences (e.g. fair testing equivalence). [7] also gives a metalanguage for concurrent strategies along with an operational semantics that exactly corresponds to its interpretation.

Hirschowitz *et al* [8] have *uncovered* models for message-passing concurrency (CCS, π) where plays are string diagrams, sound up to weak bisimulation. However, they do not form a category up to weak bisimulation, and do not consider hiding.

a) Outline: We begin Section II by introducing a simple higher-order shared-memory concurrent language, **aIPA**. To set the stage, we give **aIPA** an angelic interpretation (very close to [3]) in the category **CG** built in [14] with strategies up to isomorphism, and outline our two new interpretations, detailed in the next two sections. In Section III, we give a *non-angelic, uncovered* interpretation, with strategies up to weak bisimulation. Finally, in Section IV, we give a *non-angelic, partially covered* interpretation with strategies up to isomorphism, which is weakly bisimilar to the previous one.

II. Three interpretations of affine IPA

Idealized Parallel Algol (IPA) [9] is a toy language embodying the paradigms of higher-order shared memory concurrency. To ease the presentation of our techniques, terms shall be restricted by the type system to being *affine*: affine terms have already been used in [3], and non-affine terms could be dealt with using the techniques of *thin concurrent games* [6].

A. Syntax of aIPA

More formally, **aIPA** is an extension of the affine λ-calculus with ground state and parallel composition.

Definition II.1. *The **types** of affine IPA are* $A, B ::= \mathbb{B} \mid \mathbf{com} \mid A \multimap B \mid \mathbf{ref}_r \mid \mathbf{ref}_w$. *The notation* $\mathbb{X}$ *ranges over **ground types**,* $\mathbb{X} ::= \mathbb{B} \mid \mathbf{com}$, *which are booleans or commands. As well as a linear function space, there are types* $\mathbf{ref}_r$ *and* $\mathbf{ref}_w$ *for read-only and write-only variables (splitting* $\mathbf{ref}$ *allows the variables to be non-trivial in an affine setting).*

*The **terms** of affine IPA are the following:*

$$
\begin{array}{rcl}
b & ::= & \mathtt{tt} \mid \mathtt{ff} \\
M, N & ::= & x \mid M\,N \mid \lambda x.\,M \mid \mathtt{tt} \mid \mathtt{ff} \mid \mathtt{if}\ M\,N_1\,N_2 \mid \bot \\
& \mid & \mathbf{skip} \mid M; N \mid \mathbf{newref}\, r := b\, \mathbf{in}\, M \mid M := \mathtt{tt} \\
& \mid & !M \mid M \parallel N
\end{array}
$$

References declared by $\mathbf{newref}\, r := b\, \mathbf{in}$ are considered initialized to the constant b; having explicit initialization values is useful when defining the operational semantics. Throughout the paper, we use $\mathbf{newref}\, r\, \mathbf{in}$ as a shorthand for $\mathbf{newref}\, r := \mathtt{ff}\, \mathbf{in}$. As references can only be read once, it is only enough to be able to write one possible value ($\mathtt{tt}$ was chosen here), hence the restricted assignment command.

The typing rules are standard so we only mention a few. Firstly, affine function application and boolean elimination.

$$\frac{\Gamma \vdash M : A \multimap B \qquad \Delta \vdash N : A}{\Gamma, \Delta \vdash M\,N : B}$$

$$\frac{\Gamma \vdash M : \mathbb{B} \qquad \Delta \vdash N_1 : \mathbb{B} \qquad \Delta \vdash N_2 : \mathbb{B}}{\Gamma, \Delta \vdash \mathtt{if}\ M\,N_1\,N_2 : \mathbb{B}}$$

The first rule treats the context multiplicatively, making the language affine: it requires that Γ and Δ are partial functions from variables to types with disjoint domains of definition. For reference manipulation, we have:

$$\frac{\Gamma, r_r : \mathbf{ref}_r, r_w : \mathbf{ref}_w \vdash M : \mathbb{B}}{\Gamma \vdash \mathbf{newref}\, r := b\, \mathbf{in}\, M[r/r_w, r/r_r] : \mathbb{B}} \qquad \frac{\Gamma \vdash M : \mathbf{ref}_r}{\Gamma \vdash !M : \mathbb{B}}$$

$$\frac{\Gamma \vdash M : \mathbf{ref}_w}{\Gamma \vdash M := \mathtt{tt} : \mathbf{com}}$$

Splitting between the read and write capabilities of the variable type is necessary for the variables to be used in a non-trivial way. For example, the following term is typable:

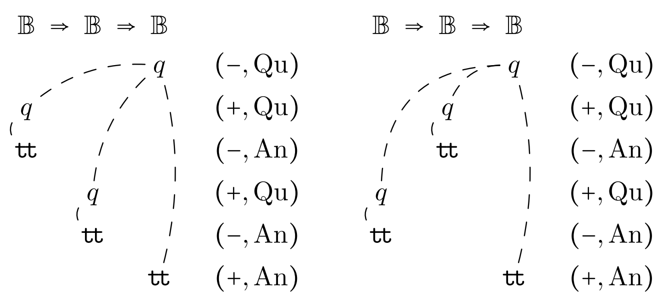
$$\begin{array}{l}\mathbf{strict} = \lambda f^{\mathbf{com} \multimap \mathbf{com}}.\, \mathbf{newref}\, r\, \mathbf{in}\, (f\,(r := \mathtt{tt}));\, !r \\ \qquad : (\mathbf{com} \multimap \mathbf{com}) \multimap \mathbb{B}\end{array}$$

Operational semantics is defined along the lines of [9] as a judgment $\Sigma \vdash M, s \to M', s'$ where Σ is a set of memory cell names, M and M' terms and s, s' are *stores*: maps $\Sigma \to \mathbb{N}$. The rules for state and parallel composition are given in Figure 1. Note that $\bot$ reduces to itself so it is an *active* divergence. A closed term of ground type $\vdash M : \mathbf{com}$ **may converge** when $\varnothing \vdash M, \varnothing \to^* \mathbf{skip}, \varnothing$. It **must converge** if it has no infinite reduction sequence.

B. Game semantics

a) An interactive semantics: Game semantics represents a program as a presentation of its possible *interactions* against a certain class of contexts. By carefully choosing the class of contexts, game semantics is very effective in capturing observational equivalence of programs for a variety of programming features [11], [1], [2], [12]. In traditional game semantics, the interaction of a program and a context is represented as a dialogue respecting the rules of a 2-player game derived from the type of the program.

For instance, the following dialogues represent the interaction of two implementations of and against a context evaluating them on true and true:



Here, dialogues are alternating Program-Context sequences of moves. In both dialogues depicted above, the context (denoted by the negative polarity) starts the dialogue by an initial question asking for the value of the computation. Then both dialogues are similar: the program (denoted by the positive polarity) asks a question representing the interrogation of an argument; then the context answers with a value; the program asks another question, on a different part of the type; the context answers it; and finally the program answers the initial question. The difference between both dialogues is the order of the questions: the leftmost dialogue starts with a question on the leftmost argument while the rightmost dialogue starts with a question on the rightmost argument. The leftmost dialogue is a possible dialogue for the left-strict and and the rightmost dialogue is a possible dialogue for the right-strict and. The dashed lines (−−) are *justification pointers*, representing the lexical scope of the calls. Game semantics interprets a term as a set of such dialogues, covering all possible behaviours under a class of contexts.

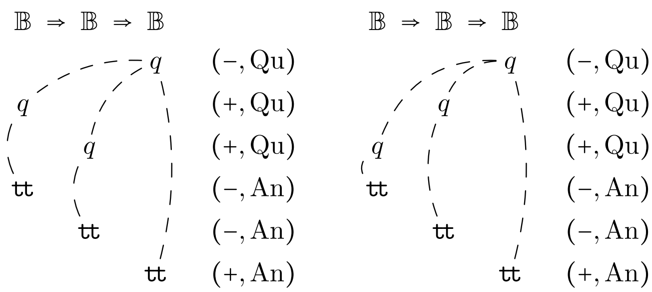
$$\frac{\Sigma \vdash M_1, s \to M_1', s'}{\Sigma \vdash (M_1 \parallel M_2), s \to (M_1' \parallel M_2), s'} \qquad \frac{\Sigma \vdash M_2, s \to M_2', s'}{\Sigma \vdash (M_1 \parallel M_2), s \to (M_1 \parallel M_2'), s'}$$

$$\frac{\Sigma, r \vdash M, s \otimes (r \mapsto b) \to M', s' \qquad r \notin M'}{\Sigma \vdash \mathbf{newref}\, r := b\, \mathbf{in}\, M, s \to M', s'} \qquad \frac{\Sigma, r \vdash M, s \otimes (r \mapsto b) \to M', s' \otimes (r \mapsto b')}{\Sigma \vdash \mathbf{newref}\, r := b\, \mathbf{in}\, M, s \to \mathbf{newref}\, r := b'\, \mathbf{in}\, M', s'}$$

$$\frac{}{\Sigma \vdash r := \mathtt{tt}, s \otimes (r \mapsto b) \to \mathbf{skip}, s \otimes (r \mapsto \mathtt{tt})} \qquad \frac{}{\Sigma \vdash !r, s \otimes (r \mapsto b) \to b, s \otimes (r \mapsto b)} \qquad \frac{}{\Sigma \vdash \bot, s \to, \bot, s}$$

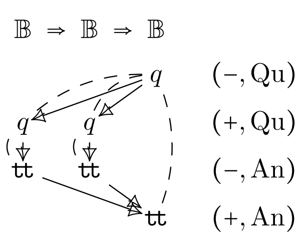
Fig. 1. Operational semantics for **aIPA**

b) Concurrent game semantics: The dialogues of the previous section consisted in *alternating* sequences of moves. At each instant, only one agent can play a move, in doing so giving control to the other. In particular, there is no way of playing two moves in a row. To model concurrent programs [13], [9], it is necessary to alleviate this constraint by allowing dialogues to be non-alternating. For instance, the dialogues below display some *concurrency*:



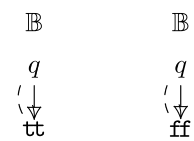
In this context, concurrency is represented by the ability of Player to ask two questions right after the other, without waiting for an answer. This is possible *e.g.* for a *parallel* implementation of and, one that evaluates both arguments in parallel and waits for both answers before returning.

c) Causal game semantics: In the previous example, the two dialogues describe the same interaction of the same program against the same context, so what is the difference between them? The only difference is the order in which the two Player questions are scheduled. Because of the sequential nature of the representation, a non-alternating dialogue displays the behaviour of a (possibly parallel) program against a (possibly parallel) context, with a choice of scheduling for the parallelism. Including explicitly the scheduling in the diagram is cumbersome. The interpretation becomes subject to a combinatorial explosion, and some intensional information about the program is lost [3]. So, we adopt instead a representation of dialogues with parallelism:

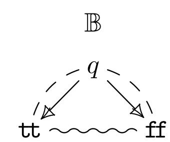


In this picture, the implicit chronological linear order is replaced by an explicit partial-order representing **causality**. Concurrency is represented by moves being incomparable (as the two player questions). In [5], we have shown how deterministic pure functional parallel programs can be interpreted using such representations.

d) Partial-orders and non-determinism: In this setting, it is easy to represent a nondeterministic program as a set of partial-orders representing the possible dialogues against concurrent contexts, as in [3]. For instance, the nondeterministic boolean would be represented as the collection:



This representation is convenient, but suffers from two drawbacks: firstly it forgets the point of non-deterministic branching. This also makes it space-inefficient, since sharing between dialogues is not represented. Secondly, one cannot talk of an *occurrence* of a move independently of an execution. Those issues can be solved by moving to *event structures* [15], where the nondeterministic boolean can be represented as:



The wiggly line ($\sim\sim$) indicates *conflict*: the two boolean values cannot coexist in an execution. This combination of causality and conflict is formalized by *event structures*:

Definition II.2. *An* ***event structure*** *is a triple* $(E, \leq_E, \mathrm{Con}_E)$ *where* $(E, \leq_E)$ *is a partial-order and* Con_E *is a non-empty collection of finite subsets of* E *called* consistent sets *subject to the following axioms:*

- *If* $e \in E$*, the set* $[e] = \{e' \in E \mid e' \leq e\}$ *is finite*
- *A subset of a consistent set is consistent,*
- *If* $X \in \mathrm{Con}_E$ *and* $e \leq e' \in X$ *then* $X \cup \{e\}$ *is consistent.*

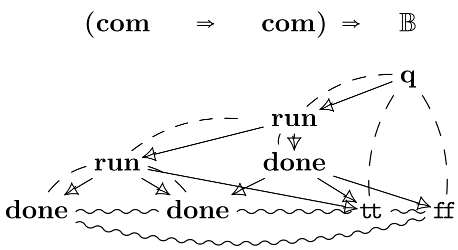
These event structures are based on *consistent sets* rather than the more commonly-encountered binary *conflict* relation. Consistent sets are more general, and more handy mathematically, but as far as diagrams are concerned, we will simply draw the Hasse diagram of $\leq$ (represented by $e \rightarrow e'$, indicating that e is an **immediate cause** of e'), along with a binary relation $\sim\sim$ of **minimal conflict** from which the consistent sets are recovered by letting $X \in \mathrm{Con}_E$ iff $\neg(e \sim\sim e')$ for all $e, e' \in [X] = \{e \in E \mid e \leq e' \in X\}$. A down-closed subset of events whose finite subsets are all consistent is called a **configuration**. The set of finite configurations of E is denoted $\mathscr{C}(E)$. If $x \in \mathscr{C}(E)$ and $e \notin x$, we write $x \overset{e}{\longrightarrow\!\!\subset} x'$ when $x' = x \cup \{e\} \in \mathscr{C}(E)$; this is the **covering relation** between configurations, and we say that e gives an **extension** of x.

All the partially ordered diagrams above denote event structures. To make these entirely formal, the missing ingredients are the *names* accompanying the events ($\mathbf{q}, \mathtt{tt}, \mathtt{ff}, \dots$) and the dashed arrows. These will come as annotations by *games*, to be introduced later, which are themselves event structures representing the types.

C. Interpretations of affine IPA with event structures

Keeping, for now, the connection with types informal, let us introduce our interpretations by showing which event structure they associate to certain terms of **aIPA**.

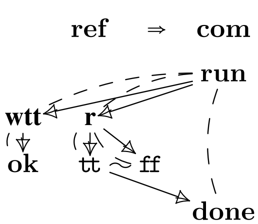
1) Angelic interpretation: In [3], we described an interpretation of **aIPA** in terms of sets of partial-orders. This interpretation can be refined in terms of event structures. For instance, the term $\mathbf{strict}$ has the following interpretation:



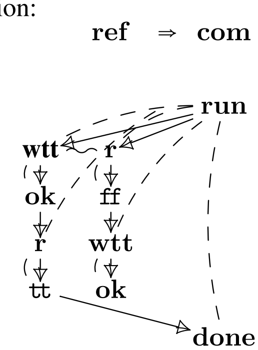
This sums up the causal structure of $\mathbf{strict}$: it returns true only if its argument calls its argument, but may return false even then, if Opponent plays both **run** and **done** concurrently.

As mentioned, this interpretation forgets hidden divergences: for instance the interpretation of $D = \mathbf{newref}\, r\, \mathbf{in}(r := \mathtt{tt} \parallel \mathtt{if}\ !r\ \mathbf{skip}\ \bot)$ is $\mathbf{run} \rightarrow \mathbf{done}$ (a strategy on $\mathbf{com}$), which does not account for the fact that D might diverge.

2) Non-angelic uncovered interpretation: The loss of divergence in the example above is due to the way composition is defined, and in particular to *hiding*. Indeed, the interpretation of the term above is obtained by first computing that of $r : \mathbf{ref} \vdash (r := \mathtt{tt} \parallel \mathtt{if}\,!r\,\mathbf{skip}\,\bot)$:

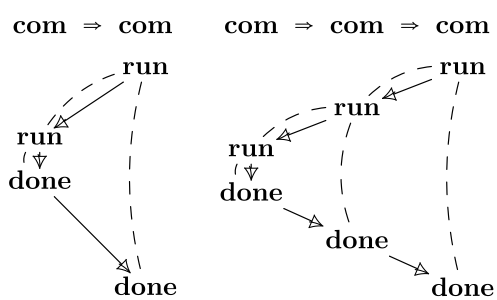


As is common in game semantics, terms containing free variables of type $\mathbf{ref}$ treat those references as uninterpreted, and operations on them, **r** for reading and **wtt** for assigning true, are simply passed on to Opponent. To compute the final semantics of D, we precompose this strategy by a strategy implementing a reference (following [2]), resulting in the following interaction:



We later describe how the interaction is obtained, but, intuitively, it synchronizes the corresponding events from the two strategies and imposes a causal ordering including constraints imposed by either; events are removed when a cyclic dependency would be induced. In particular, because the memory cell implements a sequential central memory, the memory operations are now sequentialized. Of the two available causal histories on $\mathbf{ref}$, only one leads to a visible event (namely, **done**; the events named **wtt**, **r**, $\mathtt{ff}$ and $\mathtt{tt}$ are internal to the interaction). Hence, *hiding* the component of the interaction on $\mathbf{ref}$ will ignore the other one, and yields the same behaviour as $\mathbf{skip}$.

To solve this, one is tempted to simply omit the hiding step. However, this is crucial in obtaining a category: for instance, without it, the identity strategy on **com** is no longer idempotent.



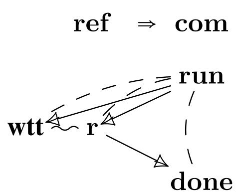
The left-hand diagram is the **copycat strategy**, interpreting the identity function, and the right-hand diagram is the interaction of copycat against itself – the **com** in the middle is shared between both copies of the copycat strategy. Hiding it yields as expected the copycat strategy back, but, *without hiding*, the interaction in itself has more events that the copycat strategy.

However, we observe that treating the events in the middle as τ-transitions, the interaction is still *weakly bisimilar* to copycat. Following these ideas, a category of *uncovered* strategies up to *weak bisimilarity* is built in Section III.

3) Interpretation with partial hiding: The issue with this solution is that considering uncovered strategies up to weak bisimulation blurs their concrete nature – *causal information* is lost, for instance. Moreover checking for weak bisimilarity is computationally expensive, and the absence of hiding increases dramatically the size of representatives: a term evaluating to **skip** may still yield a very large representative.

There is a way to cut down the strategies to reach a compromise between hiding *no* internal event, or hiding *all* of them and collapsing to an angelic interpretation.

In the setting of our games based on event structures, having a non-ambiguous notion of an occurrence of event allows us to give a very simple definition of the internal events we need to retain: those that are in a minimal conflict. This allows us to remove all internal events when composing with copycat – a necessary condition to get a category *up to isomorphism*, while still being weakly bisimilar to the uncovered strategy. The semantics of D in this setting becomes:



As before, only the events under **com** are now *visible*, *i.e.* observable by a context. But the events under **ref** are only partially hidden; those remaining are considered *internal*, treated like *τ-transitions*. Because of their presence, the partial hiding performed loses no information up to weak bisimilarity. Following these ideas, a category of partially covered strategies up to isomorphism will be constructed in Section IV.

III. Uncovered strategies up to weak bisimulation

We now construct a category of "uncovered strategies", up to weak bisimulation. Uncovered strategies are very close to the *partial strategies* of [7] – note that [7] did not aim to construct a category of partial strategies, instead focusing on connections with operational semantics.

Preliminaries on event structures: The **parallel composition** of event structures E_0 and E_1, written $E_0 \parallel E_1$ has:

- *events*: $\{0\} \times E_0 \cup \{1\} \times E_1$
- *causality*: $(i, e) \leq_{E_0 \parallel E_1} (j, e')$ when $i = j$ and $e \leq_{E_i} e'$.
- *consistent sets*: those finite subsets of $E_0 \parallel E_1$ that project to consistent sets in both E_0 and E_1

A **(partial) map of event structures** $f : A \rightharpoonup B$ is a (partial) function on events which *(1)* maps any finite configuration of A to a configuration of B, and *(2)* is locally injective: for $a, a' \in x \in \mathscr{C}(A)$ and $fa = fa'$ (both defined) then $a = a'$. In the rest of the paper, we will mainly consider total maps of event structures – hence all maps will be assumed total unless explicit mention of the contrary. We write $\mathscr{E}$ for the category of event structures and total maps and $\mathscr{E}_\perp$ for the category of event structures and partial maps.

An **event structure with partial polarities** is an event structure A with a map $pol : A \to \{-, +, *\}$ (where events are labelled "negative", "positive", or internal). It is an **event structure with total polarities** when no events are internal. A **game** is an event structure with total polarities. The dual $A^\perp$ of a game A is obtained by reversing the polarities of A. Parallel composition naturally extends to games. If x and y are configurations of an event structure with partial polarities we use $x \subseteq^p y$ where $p \in \{-, +, *\}$ for $x \subseteq y \& pol(y \setminus x) \subseteq \{p\}$.

Hiding of event structures: Given an event structure E and a subset $V \subseteq E$ of events, there is an event structure $E \downarrow V$ whose events are V and causality and consistency are inherited from E. This construction is called the **projection** of E to V and is used in [14] to perform hiding during composition.

It is sometimes convenient to work with partial maps to prove isomorphisms between event structures obtained through projection. For E and $V \subseteq E$ there is a canonical partial map $\mathrm{h} : E \rightharpoonup E \downarrow V$ defined as the (partial) identity on V. The following lemma makes reasoning on projections easy:

Lemma III.1 (Hiding maps, [4])**.** *If $f : E \rightharpoonup F$ is a partial map, the following are equivalent:*

- *Writing V for the domain of f, there is an isomorphism $\varphi : E \downarrow V \cong F$ with $\varphi \circ \mathrm{h} = f$.*
- *There exists a **hiding witness** for f that is a map $\mathrm{wit}_f : \mathscr{C}(F) \to \mathscr{C}(E)$ with $f \circ \mathrm{wit}_f(x) = x$ for $x \in \mathscr{C}(F)$ and $\mathrm{wit}_f \circ f(x) \subseteq x$ for $x \in \mathscr{C}(E)$.*

*If they hold, we say that f is a **hiding map**.*

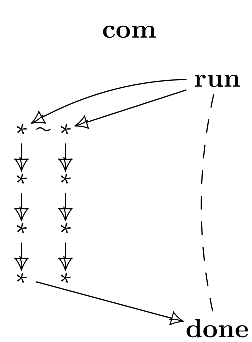
A. Definition of uncovered pre-strategies

As in [14], we start with a notion of *pre-strategies* on which composition is defined, and then refine it to a notion of strategy that behaves well with respect to copycat.

1) Uncovered pre-strategies: An uncovered pre-strategy on a game A is an event structure S partially labelled by A:

Definition III.2. *A **uncovered pre-strategy** on a game A is a total map of event structures $\sigma : S \to A \parallel N$, with N a **flat event structure**, i.e. an event structure where causality is equality and where all finite sets are consistent.*

Instead of having partial maps, a representation that makes interaction difficult to define, we have a total map that is allowed to map an event outside A, in N. An event of S is **internal** if it is mapped to N and **visible** otherwise. Although N is part of the structure of σ, it can be chosen to be arbitrarily large. This means that any finite set of strategies can be assumed (for ease of notation) to map to the same N. Uncovered pre-strategies are drawn just like the usual strategies of [14]: the event structure S has its events drawn as their labelling in A if defined or $*$ if undefined. For instance, the event structure for D given in the previous section would be drawn as:



The $--$ lines indicate **justification pointers**, but are not part of the structure. They are derived from the causalities in the game: $s - - s'$ when $\sigma s \rightarrowtriangle \sigma s'$. Here, the game is the interpretation of $\mathbf{com}$ simply described as $\mathbf{run}^- \rightarrowtriangle \mathbf{done}^+$.

From an uncovered pre-strategy, one can get a pre-strategy in the sense of [14]: for $\sigma : S \to A \parallel N$ define $S_\downarrow = S \downarrow \sigma^{-1}(A)$. By restriction σ yields $\sigma_\downarrow : S_\downarrow \to A$, called a **covered pre-strategy**.

An uncovered pre-strategy **from a game** A **to a game** B is a map $\sigma : S \to A^\perp \parallel N \parallel B$ (which up to the isomorphism $A^\perp \parallel N \parallel B \cong A^\perp \parallel B \parallel N$ can be seen as an uncovered pre-strategy on $A^\perp \parallel B$.)

As an example, we introduce the copycat pre-strategy on a game A which is the same as in [14]:

Definition III.3. *The copycat strategy on* A *is given by the mapping* $\text{æ}_A : \mathbb{C}\mathbb{C}_A \to A^\perp \parallel A$ *(N is empty here so we omit it) where* $\mathbb{C}\mathbb{C}_A$ *is the event structure defined as:*

- events*: those of* $A^\perp \parallel A$
- causality*: the transitive closure of (polarities taken in* A*)*
$$\begin{aligned} \leq_{A^\perp \parallel A} \quad & \cup \{((0,a),(1,a)) \mid a \text{ positive}\} \\ & \cup \{((1,a),(0,a)) \mid a \text{ negative}\} \end{aligned}$$
- consistent sets: *those sets* $X \subseteq A^\perp \parallel A$ *such that the down-closure of* X *inside the partial order* $\leq_{\mathbb{C}\mathbb{C}_A}$ *defined above is consistent in* $A^\perp \parallel A$.

Isomorphism of strategies introduced in [14] can be extended to uncovered pre-strategies:

Definition III.4. *Two uncovered pre-strategies* $\sigma : S \to A \parallel N$ *and* $\tau : T \to A \parallel N$ *are* ***isomorphic*** *(written* $\sigma \cong \tau$*) when there exists an iso* $\varphi : S \cong T$ *that restricts to an iso on the visible part:* $\varphi(S_\downarrow) = T_\downarrow$ *and* $\tau_\downarrow \circ \varphi = \sigma_\downarrow : S_\downarrow \to A$.

2) Interaction of pre-strategies: The interaction of uncovered pre-strategies will be described as a certain pullback in the category of event structures, following the lines of [14]. We briefly sketch the construction of such pullbacks.

Given maps of event structures $f : A \to C$, $g : B \to C$, define an **interaction state** to be a pair $(x,y) \in \mathscr{C}(A) \times \mathscr{C}(B)$ such that $fx = gy \in \mathscr{C}(C)$ and the induced bijection $\varphi_{(x,y)} : x \cong fx = gy \cong y$ is **secured**: the natural preorder on $\varphi_{(x,y)}$ defined on the graph of $\varphi_{(x,y)}$ by the transitive closure of $\{((s,t),(s',t')) \mid s < s' \vee t < t'\}$ is a partial-order.

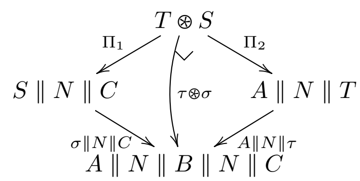
Lemma III.5. *The following is an event structure* $S \wedge T$*:*

- events*: interaction states* (x,y) *for which the partial-order* $\varphi_{(x,y)}$ *has a top-element, written* $(\Pi_1(x,y), \Pi_2(x,y))$*, where* $\Pi_1(x,y) \in S, \Pi_2(x,y) \in T$.
- causality*: given by pairwise inclusion*
- consistency*: a set of interaction states* X *is consistent if* $\left(\bigcup_{(x,y)\in X} x, \bigcup_{(x,y)\in X} y\right)$ *is an interaction state.*

Moreover the mapping $(x,y) \mapsto \Pi_1(x,y)$ *and* $(x,y) \mapsto \Pi_2(x,y)$ *define maps of event structures* $S \wedge T \to S$ *and* $S \wedge T \to T$ *such that* $(S \wedge T, \Pi_1, \Pi_2)$ *is a pullback of* f *and* g *both in* $\mathscr{E}$ *and in* $\mathscr{E}_\perp$.

The construction is explained in more detail in [4].

Given $\sigma : S \to A^\perp \parallel N \parallel B$ and $\tau : T \to B^\perp \parallel N \parallel C$ we form the following pullback that is the interaction of σ and τ:



where $T \circledast S$ is $(S \parallel N \parallel C) \wedge (A \parallel N \parallel T)$. The main difference with [14] is the addition of internal events and their treatment: they do *not* synchronize. For σ, the internal events of τ occur in the background and vice-versa. The resulting map $\tau \circledast \sigma : T \circledast S \to A \parallel N \parallel B \parallel N \parallel C$ can be viewed as an uncovered pre-strategy from A to C via $\tau \circledast \sigma : T \circledast S \to A \parallel (N \parallel B \parallel N) \parallel C$. The events sent to B become internal.

3) Weak bisimulation: To compare uncovered pre-strategies, we cannot use isomorphism as in [14], since as observed in the introduction $\text{æ}_A \circledast \sigma$ is large than σ. To solve this, we introduce weak bisimulation between uncovered strategies:

Definition III.6. *Let* $\sigma : S \to A \parallel N$ *and* $\tau : T \to A \parallel N$ *be uncovered pre-strategies. A weak bisimulation between* σ *and* τ *is a relation* $\mathscr{R} \subseteq \mathscr{C}(S) \times \mathscr{C}(T)$ *such that:*

- $(\varnothing, \varnothing) \in \mathscr{R}$
- *If* $x \overset{s}{\longrightarrow\!\!\subset} x'$ *such that* s *is visible, then there exists* $y \subseteq^* y' \overset{t}{\longrightarrow\!\!\subset} y''$ *with* $\sigma s = \tau t$ *and* $x' \mathscr{R} y''$ *(and the converse condition for* τ*)*
- *If* $x \overset{s}{\longrightarrow\!\!\subset} x'$ *such that* s *is internal, then there exists* $y \subseteq^* y'$ *such that* $x' \mathscr{R} y'$ *(and the converse condition for* τ*)*

Two uncovered pre-strategies σ, τ *are weakly bisimilar (written* $\sigma \simeq \tau$*) when there is a weak bisimulation between them.*

Lemma III.7. *Interaction is associative up to isomorphism (hence up to weak bisimulation):* $\sigma \circledast (\tau \circledast \upsilon) \cong (\sigma \circledast \tau) \circledast \upsilon$.

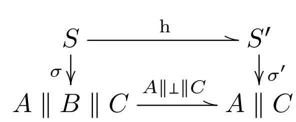
Proof. Follows from the universal property of pullbacks. □

Lemma III.8. *Weak bisimulation is a congruence with respect to interaction: if* $\sigma \simeq \sigma'$ *are weakly bisimilar uncovered pre-strategies from* A *to* B *and* τ *is an uncovered pre-strategy from* B *to* C*, then* $\tau \circledast \sigma \simeq \tau \circledast \sigma'$.

Proof. If $\mathscr{R}$ is a weak bisimulation between σ and σ' then $\tau \circledast \mathscr{R} = \{(w,z) \mid (\Pi_1 w) \mathscr{R} (\Pi_1 z) \;\&\; \Pi_2 w = \Pi_2 z\}$ is the desired weak bisimulation. □

4) Zipping lemma: The following lemma captures elegantly the interaction between interaction and hiding:

Lemma III.9 (Zipping lemma)**.** *Let* $\sigma : S \to A \parallel B \parallel C$ *and* $\sigma' : S \to A \parallel C$ *be maps of event structures. Take* $\mathrm{h} : S \rightharpoonup S'$ *be a hiding map making the following diagram commute:*



Then, for $\rho : U \to C^\perp \parallel D$, the morphism $U \circledast \mathrm{h} : U \circledast S \rightharpoonup U \circledast S'$ defined using the universal property of $U \circledast S'$ of pullbacks in $\mathscr{E}_\perp$ is a hiding map.

Proof. See [4]. □

5) Composition of covered strategies: From interaction, we can define the usual composition of covered strategies easily. If $\sigma : S \to A^\perp \parallel B$ and $\tau : T \to B^\perp \parallel C$ are covered pre-strategies, their composition (in the sense of [14]) $\tau \odot \sigma$ is defined as $(\tau \circledast \sigma)_\downarrow$. The operation $\downarrow$ is well-behaved with respect to interaction:

Lemma III.10. *Let σ, τ be composable uncovered pre-strategies. We have*

$$(\tau \circledast \sigma)_\downarrow \cong \tau_\downarrow \odot \sigma_\downarrow.$$

Proof. We apply twice the Zipping Lemma (Lemma III.9) to the hiding maps $\mathrm{h}_S : S \rightharpoonup S_\downarrow$ and $\mathrm{h}_T : T \rightharpoonup T_\downarrow$ yielding the composition of *partial* maps:

$$T \circledast S \xrightarrow{T_\downarrow \circledast \mathrm{h}_S} T \circledast S_\downarrow \xrightarrow{\mathrm{h}_T \circledast S_\downarrow} T_\downarrow \circledast S_\downarrow \to (T_\downarrow \circledast S_\downarrow)_\downarrow$$

Since all are hiding maps, the composition is also a hiding map with domain the visible events of $T \circledast S$, which gives the result by Lemma III.1. □

B. A compact-closed category of uncovered strategies

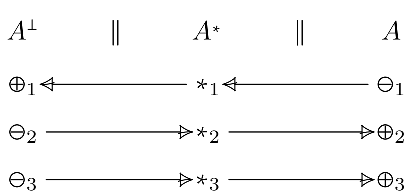
We now move on to defining a compact-closed category of uncovered strategies up to weak bisimulation. We have a tentative definition of morphisms (uncovered pre-strategies), with a composition operation which by Lemma III.7 is associative. The only missing ingredient is copycat: as in [14], we do not have $\text{æ}_A \circledast \sigma \simeq \sigma$ in general. In this section, we give conditions on pre-strategies for this to hold.

1) Race-free games and copycat: The first thing to check is that copycat is indeed idempotent. In [14], this is true, but in our setting it does not hold automatically for every game.

Consider the game $A = \ominus_1 \oplus_2 \oplus_3$ with trivial causality, and consistency given by

$$X \in \mathrm{Con}_A \text{ iff } |X| \leq 2$$

Events are pairwise compatible, but all three cannot occur together. In particular, $\{\ominus_1, \oplus_2\}$ is a maximal configuration of A. The interaction of copycat with itself is:



The conflict, as it is not binary, is *not* represented on the above picture. Any bisimulation between $\text{æ}_A \circledast \text{æ}_A$ and æ_A must relate the minimal configurations of $\text{æ}_A \circledast \text{æ}_A$ and æ_A featuring $\{\ominus_1, \ominus_2, \ominus_3\} \in \mathscr{C}(A^\perp \parallel A)$. From there, $\text{æ}_A \circledast \text{æ}_A$ can do a silent transition to $\{\ominus_1, \ominus_2, \ominus_3, *_1, *_2\}$ (and æ_A does nothing since there are no internal events in æ_A). But æ_A can perform a visible transition to $\{\ominus_1, \ominus_2, \ominus_3, \oplus_3\}$, which cannot be matched by $\text{æ}_A \circledast \text{æ}_A$, as $\{*_1, *_2\}$ is maximal in A^*.

A sufficient condition to avoid this problem is to restrict ourselves to race-free games: a game A is **race-free** when if x can be extended by two events a_1, a_2 of distinct polarities, the union $x \cup \{a_1, a_2\}$ is consistent. Race-freeness is sufficient to ensure that copycat is idempotent:

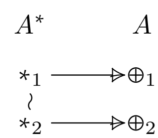
Lemma III.11. *For a race-free game A, $\text{æ}_A \circledast \text{æ}_A \simeq \text{æ}_A$.*

Proof. It will follow from the forthcoming Lemma III.13. □

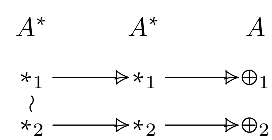
Note that race-freeness is not necessary. We believe the exact characterisation is linked to phenomena studied in [7]. In the rest of the paper we only consider race-free games.

2) Uncovered strategies: Finally, we characterise the pre-strategies invariant under composition with copycat. The two ingredients of [14], [4], receptivity and courtesy (called *innocence* in [14]) are needed, but this is not enough: we need another condition as witnessed by the following example.

Consider the strategy $\sigma : \oplus_1 \sim\sim \oplus_2$ on the game $A = \oplus_1 \oplus_2$ playing nondeterministically one of the two moves. Then the interaction $\text{æ}_A \circledast \sigma$ is:



It is not weakly bisimilar to σ: $\text{æ}_A \circledast \sigma$ can do $*_1$, an internal transition, to which σ can only respond by not doing anything. Then σ can still do $\oplus_1$ and $\oplus_2$ whereas $\text{æ}_A \circledast \sigma$ cannot: it is committed to doing $\oplus_1$. To solve this problem, we need to force strategies to decide their nondeterministic choices *secretly*, by means of internal events – so σ will not be a valid uncovered strategy, but $\text{æ}_A \circledast \sigma$ will. Indeed, $\text{æ}_A \circledast (\text{æ}_A \circledast \sigma)$ below



is indeed weakly bisimilar to $\text{æ}_A \circledast \sigma$.

Accordingly, we define uncovered strategies:

Definition III.12. *An uncovered strategy is an uncovered pre-strategy $\sigma : S \to A \parallel N$ satisfying:*

- receptivity: *if $x \in \mathscr{C}(S)$ is such that $\sigma x \xrightarrow{a}\subset$ with $a \in A$ negative, then there exists a unique $x \xrightarrow{s}\subset$ with $\sigma s = a$.*
- courtesy: *if $s \rightarrowtriangle s'$ and s is positive or s' is negative, then $\sigma s \rightarrowtriangle \sigma s'$.*
- secrecy: *if $x \in \mathscr{C}(S)$ extends with s_1, s_2 but $x \cup \{s_1, s_2\} \notin \mathscr{C}(S)$, then s_1 and s_2 are either both negative, or both internal.*

Receptivity and courtesy are stated exactly as in [14]. Unlike in [14], though we have neutral events, courtesy in our settings allows strategies to add causal links of the form $- \rightarrowtriangle +, - \rightarrowtriangle$

$*, * \rightarrowtriangle *$ and $* \rightarrowtriangle +$. As a result, hiding the internal events of an uncovered strategy yields a strategy: for an uncovered strategy $\sigma : S \to A \parallel N$, the covered pre-strategy $\sigma_\downarrow$ is a strategy in the sense of [14].

For any game A, æ_A is an uncovered strategy: it satisfies secrecy since the only minimal conflicts it has are inherited from the game and are between negative events.

3) The category $\mathbf{CG}_\circledast$: Our definition of uncovered strategy does imply that copycat is neutral for composition.

Lemma III.13. *Let $\sigma : S \to A \parallel N$ be an uncovered strategy. Then $\text{æ}_A \circledast \sigma \simeq \sigma$.*

Proof. The weak bisimulation is given by:

$$\mathscr{R} = \{(x, z) \in \mathscr{C}(S) \times \mathscr{C}(\mathbb{C}_A \circledast S) \mid x \sqsubseteq_S \Pi_1 z \,\&\, \sigma x = \Pi_2 z\}$$

where the Scott order $x \sqsubseteq_S x'$ is $x \supseteq^- x \cap x' \subseteq^+ x'$. That it is a weak bisimulation requires a bit of work, but is a simplification of the proof that copycat is neutral for composition [4]. □

The result follows immediately:

Theorem III.14. *The following data defines a compact-closed category $\mathbf{CG}_\circledast$ up to weak bisimulation:*

- objects*: race-free games,*
- maps from A to B*: uncovered strategies $\sigma : S \to A^\perp \parallel N \parallel B$ from A to B, up to weak bisimulation.*

The tensor product is given by parallel composition of games, and the dual operation by the duality on games.

Proof. The fact that we have a category up to weak bisimulation follows from Lemmata III.8, III.7, III.13. The compact-closed structure follows closely the proof of [4] by lifting the structural morphisms for the monoidal structure of parallel composition in $\mathscr{E}$ to strategies. □

C. Interpretation of affine IPA

From now on all strategies are by default considered uncovered, unless stated otherwise explicitly. We end this section by sketching the interpretation of affine IPA inside $\mathbf{CG}_\circledast$. As a compact-closed category, $\mathbf{CG}_\circledast$ supports an interpretation of the linear λ-calculus. However, the unit for the tensor product (the empty game) is not terminal. As a result, there is no natural transformation $\epsilon_A : A \to 1$ in $\mathbf{CG}_\circledast$.

1) The negative category $\mathbf{CG}^-_\circledast$: We solve this issue as in [3], by looking at negative strategies and negative games.

Definition III.15. *An event structure with partial polarity is* ***negative*** *when all its minimal events are negative.*

A strategy $\sigma : S \to A \parallel N$ is negative when S is.

Copycat on a negative game is negative, and negative strategies are stable under composition:

Lemma III.16. *There is a subcategory $\mathbf{CG}^-_\circledast$ of $\mathbf{CG}_\circledast$ consisting in negative race-free games and negative strategies. It inherits a monoidal structure from* **CG** *in which the unit (the empty game) is terminal.*

Besides a terminal object, $\mathbf{CG}^-_\circledast$ has *products*. For two games A and B, their **product** $A \,\&\, B$ has events, causality, polarities as for $A \parallel B$, but consistent sets restricted to those whose projection to either A or B is empty. The **projections** are

$$\varpi_A : \mathbb{C}_A \to (A \,\&\, B)^\perp \parallel A \qquad \varpi_B : \mathbb{C}_B \to (A \,\&\, B)^\perp \parallel B$$

Finally, the **pairing** of negative strategies $\sigma : S \to A^\perp \parallel N \parallel B$ and $\tau : T \to A^\perp \parallel N \parallel C$ is the obvious map

$$\langle \sigma, \tau \rangle : S \,\&\, T \to A^\perp \parallel N \parallel N \parallel B \,\&\, C,$$

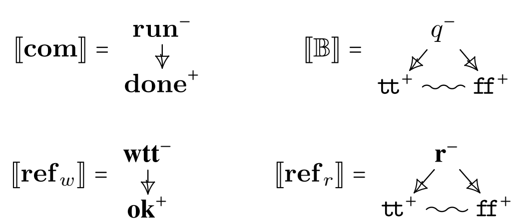
and the laws for the cartesian product are direct verifications.

We also need a construction to interpret the function space. However, $\mathbf{CG}^-_\circledast$ does not inherit the closed structure of $\mathbf{CG}_\circledast$: for A and B negative, $A^\perp \parallel B$ is not usually negative. To circumvent this, we introduce the linear arrow $A \multimap B$, a negative version of $A^\perp \parallel B$. To simplify the presentation, we only define it in a special case. A game is **well-opened** when it has at most one initial event. When B is well-opened, we define $A \multimap B$ to be 1 if $B = 1$; and otherwise $A^\perp \parallel B$ with the exception that every move in A depends on the single minimal move in B. As a result $\multimap$ preserves negativity. We get:

Lemma III.17. *If B is well-opened, there is an identity between:*

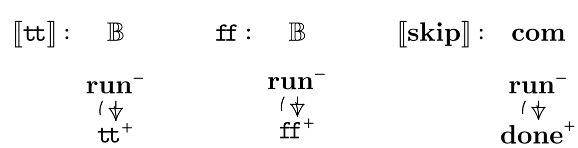
- *Negative strategies $\sigma : S \to A^\perp \parallel N \parallel (B^\perp \parallel C)$*
- *Negative strategies $\sigma : S \to A^\perp \parallel N \parallel (B \multimap C)$*

The games $B \multimap C$ and $B^\perp \parallel C$ have the same events (for C non-empty), so this identity comes from the fact that by negativity, any strategy $\sigma : S \to A^\perp \parallel N \parallel (B^\perp \parallel C)$ automatically type-checks as $\sigma : S \to A^\perp \parallel N \parallel (B \multimap C)$. From this, and leveraging the compact closed structure of $\mathbf{CG}_\circledast$, it is elementary to prove that $A \multimap B$ is an exponential object of A and B in $\mathbf{CG}^-_\circledast$ – and it is still well-opened. In other words, well-opened games are an exponential ideal in $\mathbf{CG}^-_\circledast$. This gives us directly the interpretation of types of **aIPA** inside well-opened games of $\mathbf{CG}^-_\circledast$:



$$[\![A \multimap B]\!] = [\![A]\!] \multimap [\![B]\!]$$

2) Interpretation of terms: Interpretation of the affine λ-calculus in $\mathbf{CG}^\circledast_-$ follows standard methods. First, the constants $\mathtt{tt}, \mathtt{ff}, \mathbf{skip}$ are interpreted as:



The strategies implementing **aIPA** constructs are given in Figure 2. The semantics is obtained by postcomposing with these strategies:

$$\begin{aligned}
[\![M;N]\!]_\circledast &= \mathtt{seq} \circledast ([\![M]\!]_\circledast \parallel [\![N]\!]_\circledast) \\
[\![M \coloneqq \mathtt{tt}]\!]_\circledast &= \mathtt{write} \circledast [\![M]\!]_\circledast \\
[\![M \parallel N]\!]_\circledast &= \mathtt{join} \circledast ([\![M]\!]_\circledast \parallel [\![N]\!]_\circledast) \\
[\![!M]\!]_\circledast &= \mathtt{read} \circledast [\![M]\!]_\circledast \\
[\![\mathtt{if}\ M\,N\,N']\!]_\circledast &= \mathtt{if} \circledast ([\![M]\!]_\circledast \parallel \langle [\![N]\!]_\circledast, [\![N']\!]_\circledast \rangle) \\
[\![\mathbf{newref}\, r \coloneqq b\,\mathbf{in}\, M]\!]_\circledast &= [\![M]\!]_\circledast \circledast \mathbf{cell}_b
\end{aligned}$$

A non-standard point is the interpretation of $\bot$: usually interpreted in game semantics by the minimal strategy simply playing q (as will be done in the next section), our interpretation here reflects the fact that $\bot$ represents an infinite computation that never returns. Otherwise, this interpretation follows very closely the lines of [3]. In particular, references are implemented by precomposing with a sequential memory **cell**. More precisely, given a term $\Gamma, r_r : \mathbf{ref}_r, r_w : \mathbf{ref}_w \vdash M : \mathbb{B}$, the interpretation of $\mathbf{newref}\, r \coloneqq b\,\mathbf{in}\, M$ is defined as $[\![M]\!]_\circledast \circledast \mathbf{cell}_b$, where $[\![M]\!]_\circledast$ is viewed as a strategy from $\mathbf{ref}_r \parallel \mathbf{ref}_w$ to $[\![\Gamma]\!]^\perp \parallel \mathbb{B}$ by curryfication. This is indeed well-defined even though $\mathbf{cell}_{\mathtt{ff}}$ does not satisfy any of the conditions for uncovered strategies:

Lemma III.18. *For any uncovered strategy $\sigma : S \to A^\perp \parallel \mathbf{ref}_r^\perp \parallel \mathbf{ref}_w^\perp \parallel \mathbb{B}$, the pre-strategy $\sigma \circledast (\,\text{\c{c}}_A \parallel \mathbf{cell}_{\mathtt{ff}})$ is an uncovered strategy.*

Proof. Direct verification of the axioms. □

Since our language is finite, there are only two possible complete interaction traces on variable: either the term reads then writes or writes then reads. The pre-strategy $\mathbf{cell}_b$ chooses nondeterministically between those two possibilities.

3) A sound and adequate interpretation: We now prove that our interpretation $[\![\cdot]\!]_\circledast$ is sound and adequate for may and must convergences. This means that a term may (resp. must) converge if and only if its interpretation may (resp. must) converge. However, we have not defined what it means for a strategy to may or must converge. May convergence is easy: an uncovered strategy σ on **com** may converge if the only positive move of $[\![\mathbf{com}]\!]$ is in the image of σ. Must convergence is less obvious to define. We follow [7]:

Definition III.19. *A strategy σ on* **com** ***must converge*** *if all configurations maximal for inclusion contain a positive move.*

We can see that $[\![D]\!]_\circledast$ must not converge since, once the left internal event is performed, no positive move can ever be played. This abstract definition has a very concrete understanding in the image of the interpretation:

Lemma III.20. *For a term $\vdash M : \mathbb{B}$, $[\![M]\!]$ must converge if and only if $[\![M]\!]$ does not have infinite configurations.*

Proof. This is done by proving that strategies in the image of the interpretation satisfy the following two properties:

1) If two positive events s, s' are concurrent but not bounded (that is there a common event above s and s'), then there exists two negative events s_0, s_0' that are either minimal or with the same predecessor, such that $s_0 < s$ and $s_0' < s'$.
2) Internal events are never maximal.

The first property encodes the fact that concurrency in **aIPA** is always *joined*: one cannot start a thread and ignore its return value. The second assumption means that a strategy corresponding to a term never stops computing without yielding a value. Those two properties can be checked to hold for building blocks of the interpretation and be stable under composition, so they hold for the whole interpretation. The equivalence follows from these two invariants. □

This corresponds nicely with the syntactic notion of must-convergence as having no infinite runs. Both notions of convergence are well-behaved with respect to weak bisimulation:

Lemma III.21. *Let $\sigma : S \to A \parallel N, \tau : T \to A \parallel N$ be uncovered strategies on* **com**. *If $\sigma \simeq \tau$ and σ may (resp. must) converge then τ may (resp. must) converge.*

Proof. Let $\mathscr{R}$ be a bisimulation between σ and τ. By induction, we can build a map $f : \mathscr{C}(S) \to \mathscr{C}(T)$ such that x and $f(x)$ are related by $\mathscr{R}$ and have same image in A, and similarly $g : \mathscr{C}(T) \to \mathscr{C}(S)$ satisfying the corresponding assumptions.

If σ may converge, there exists a configuration $x \in \mathscr{C}(S)$ with a positive move. Then fx is a configuration of T with a positive move. Assume σ must converge. Let $y \in \mathscr{C}(T)$. By assumption, $gy \mathscr{R} y$ and gy must extend to x' with a positive move. By applying the bisimulation rules, we find that y must extend to y' such that x' and y' have the same projection to A so in particular y' has a positive move. □

To prove adequacy of this interpretation, we first prove a correspondence between the denotational and operational semantics:

Lemma III.22. *Let $\Gamma \vdash M : A$ be a term and let $\mathscr{M} = \{M' \mid \Sigma \vdash M, s \to M', s'\}$ be the set of all possible reducts of M. Then $[\![M]\!]$ has an infinite configuration if and only if there exists $M' \in \mathscr{M}$ such that $[\![M']\!]$ has an infinite configuration.*

Proof. By induction on M. □

Corollary III.23. *Let M be a closed term of type* **com** *and $\mathscr{M} = \{M' \mid \Sigma \vdash M, \varnothing \to M', \varnothing\}$. The strategy $[\![M]\!]_\circledast$ must converge if and only if all the $[\![M']\!]$ must converge*

Proof. By applying Lemma III.22 together with Lemma III.21. □

We could get a stronger link between the two denotations, but this is enough to prove adequacy:

Theorem III.24. *The interpretation $[\![\cdot]\!]_\circledast$ is sound and adequate for may and must convergence, meaning:*

1) *A term $\vdash M : \mathbb{X}$ may converge if and only if $[\![M]\!]_\circledast$ contains a positive move*

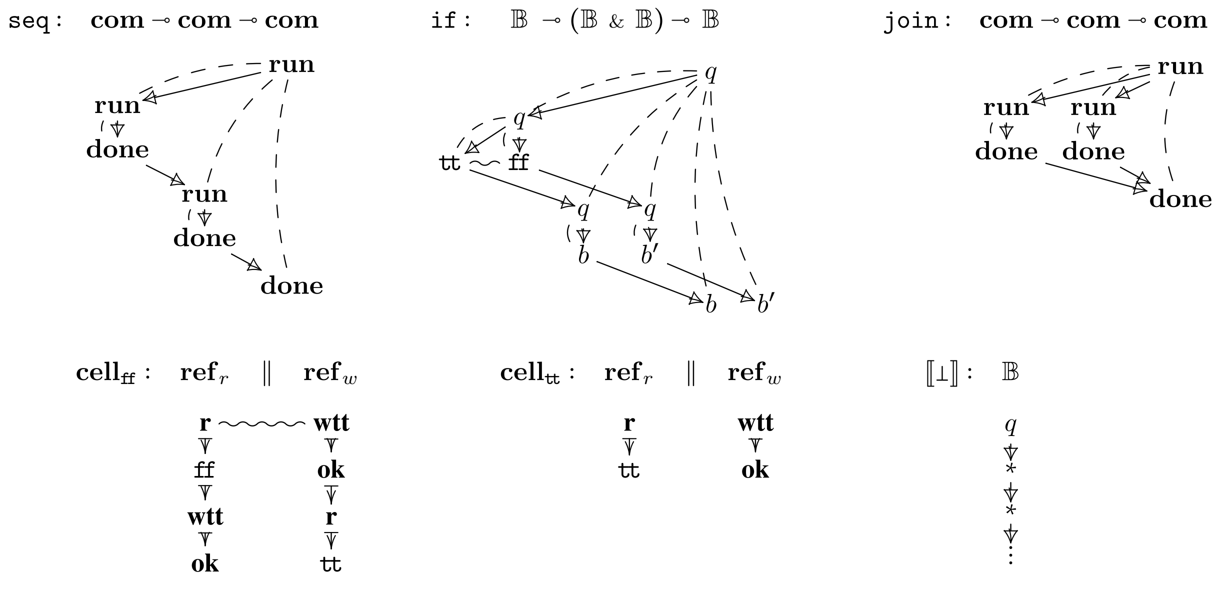


Fig. 2. Interpretation of the constructions of **aIPA**

2) *A term $\vdash M : \mathbb{X}$ must converge if and only if $[\![M]\!]_{\circledast}$ must converge.*

Proof. May-convergence. An uncovered σ contains a positive move if and only if $\sigma_{\downarrow}$ does. As a result, we can then leverage the result of [3] where the partial-order model is shown to be sound and adequate by collapsing to the model of [9].

Must-convergence. Because our language is affine, if M must converge, then there exists a global bound on the length of any reduction path of M that we write $\nu(M)$. By induction on $\nu(M)$, we prove that $[\![M]\!]_{\circledast}$ must converge. If $\nu(M) = 0$, then M is $\mathtt{tt}$ or $\mathtt{ff}$ which must converge. Otherwise, by induction hypothesis we have that all one-step reducts of $[\![M]\!]_{\circledast}$ must converge, and by Corollary III.23, so must $[\![M]\!]_{\circledast}$.

Conversely, assume $[\![M]\!]_{\circledast}$ must converge, but M must not converge. This means that $\varnothing \vdash M, \varnothing \to^{*} \bot, \varnothing$, which contradicts the first statement of Corollary III.23. □

This argues for the games model providing a good compositional LTS-semantics, one with independence built in.

IV. Essential events

The model presented in the previous section is very operational: configurations of $[\![M]\!]_{\circledast}$ can be seen as derivations for an operational semantics. The price, however, is that besides the fact that the interpretation grows dramatically in size, we can only get a category up to weak bisimulation.

We wish now to forget most of the information that is not relevant to characterise the behaviour of terms up to weak bisimulation. In other words, we want a notion of *essential internal events* that *(1)* is conservative with respect to weak bisimulation (forgetting inessential events does not lose behaviours), but, *(2)* enough events are forgotten to get a category up to isomorphism (which amounts to $\text{æ}_A \circ \sigma \cong \sigma$).

A. Definition of essential events

As illustrated before, the loss of behaviours when hiding is due to the disappearance of events participating in a conflict. A neutral event may not have visible consequences but still be relevant if in a minimal conflict; we will then call it *essential*.

However, our previously introduced notion $\sim\!\sim$ of minimal conflict is not sufficient. Indeed our event structures carry an arbitrary consistency relation rather than binary conflict, so relevant minimal conflicts may be *contextual*. Two extensions e and e' of x are **compatible** when $x \cup \{e, e'\} \in \mathscr{C}(E)$, **incompatible** otherwise. In the latter case, we have a **minimal conflict** between e and e' **in context** x (written $e \sim\!\sim_x e'$).

We can now define essential events.

Definition IV.1. *Let $\sigma : S \to A \parallel N$ be an uncovered pre-strategy. An **essential event** of S is an event s which is either visible, or (internal and) involved in a minimal conflict (that is such that we have $s \sim\!\sim_x s'$ for some s', x.)*

We write E_S for the set of essential events of σ. Any uncovered pre-strategy $\sigma : S \to A \parallel N$ induces another uncovered pre-strategy $\mathscr{E}(\sigma) : \mathscr{E}(S) = S \downarrow E_S \to A \parallel N$ called **the essential part** of σ.

The following proves that our definition satisfies *(1)*: no behaviour is lost.

Lemma IV.2. *An uncovered pre-strategy $\sigma : S \to A \parallel N$ is weakly bisimilar to its essential part.*

Proof. Define $\mathscr{R} = \{(x, x \cap E_S) \mid x \in \mathscr{C}(S)\}$. We prove it is a weak bisimulation. Let $(x, x \cap E_S) \in \mathscr{R}$.
First, assume x can extend by an event s. Either $s \in E_S$ and then $x \cap E_S$ also extends by s in $S \downarrow E_S$ and $(x \cup \{s\}, x \cap E_S \cup \{s\}) \in \mathscr{R}$ as desired, or $s \notin E_S$ (then s is internal) and $(x \cup \{s\}, x \cap E_S) \in \mathscr{R}$.

Now assume $x \cap E_S$ can be extended by an event $s \in E_S$.

Assume that $x \cup [s] \notin \mathscr{C}(S)$: since it is down-closed, it must be that $x \cup [s] \notin \mathrm{Con}_S$. Since x and $[s]$ are consistent, there must exist $x' \subseteq x \cup [s]$ with two incompatible extensions $s_0 \in x$ and $s_1 \in [s]$. By definition, s_0 and s_1 are essential. This is absurd because this would mean that $(x' \cap E_S) \cup \{s_0, s\} \subseteq x \cap E_S \cup \{s\} \in \mathscr{C}(S \downarrow E_S)$ but this set is not consistent in S (hence not consistent in $S \downarrow E_S$).

Having just proved that $x \cup [s] \in \mathscr{C}(S)$, the conclusion follows as $(x \cup [s], \{s\} \cup (x \cap E_S)) \in \mathscr{R}$. □

This induces a new notion of composition that only keeps the essential events. For $\sigma : S \to A^\perp \parallel N \parallel B$ and $\tau : T \to B^\perp \parallel N \parallel C$ we define $\tau \circledcirc \sigma = \mathscr{E}(\tau \circledast \sigma)$.

Lemma IV.3. *Operator $\circledcirc$ is associative up to isomorphism.*

Moreover, $\mathscr{E}$ behaves well with respect to composition:

Lemma IV.4. *Let σ and τ be composable uncovered pre-strategies. We have $\mathscr{E}(\tau \circledast \sigma) \cong \mathscr{E}(\tau) \circledcirc \mathscr{E}(\sigma)$.*

Proof. The proof goes as for Lemma III.10. □

We now show how to recover a category up to isomorphism by considering some uncovered strategies.

B. The category $\mathbf{CG}_\circledcirc$

In this subsection, we build a category $\mathbf{CG}_\circledcirc$ out of some uncovered strategies up to isomorphism, proving property (2): events arising in the composition with copycat are inessential.

To do so, we study the essential events of $\text{Ⅽ}_A \circledcirc \sigma$.

1) Essential events of $\text{Ⅽ}_A \circledcirc \sigma$: Let $\sigma : S \to A \parallel N$ be an uncovered strategy. We study the composition $\text{Ⅽ}_A \circledcirc \sigma$. Remember that $\mathbb{C}_A \circledast S$ is defined as the pullback of $\sigma \parallel A : S \parallel A \to N \parallel A \parallel A$ against $N \parallel \text{Ⅽ}_A : N \parallel \mathbb{C}_A \to N \parallel A \parallel A$, and the composition $\mathbb{C}_A \circledcirc S$ is defined as the projection of $\mathbb{C}_A \circledast S$ to essential events.

First, because of secrecy, internal essential events of a composition arise from those of the composed strategies:

Lemma IV.5. *Let $\sigma : S \to A^\perp \parallel N \parallel B$ and $\tau : T \to B^\perp \parallel N \parallel C$ be uncovered strategies. An internal event p of $T \circledcirc S$ is essential iff $\Pi_1 p \in S \parallel N \parallel C$ is internal and essential in S, or $\Pi_2 p \in A \parallel N \parallel T$ is internal and essential in T.*

Proof. First, we prove it for the interaction $T \circledast S$. It is an elementary lemma in concurrent games (see *e.g.* [4]) that a minimal conflict in $T \circledast S$ projects to a minimal conflict to either S or T. Moreover, the corresponding projection must be internal (and hence essential) since by secrecy there are no minimal conflict between positive moves in an uncovered strategy. Dually, an internal minimal conflict in either S or T yields immediately a minimal conflict in $T \circledast S$. To deduce the result for $T \circledcirc S$, we simply notice that by definition of essential events, no events involved in a minimal conflict are hidden when constructing $T \circledcirc S$. □

This yields a characterisation of the internal essential events of the composition $\mathbb{C}_A \circledcirc S$. Given a configuration $x \in \mathscr{C}(S)$, we write $\widehat{x} = \sigma_N(x) \parallel \sigma_\downarrow(x_\downarrow) \parallel \sigma_\downarrow(x_\downarrow) \in \mathscr{C}(N \parallel \mathbb{C}_A)$ where $\sigma_N : S \rightharpoonup N$ is the obvious partial map. We have:

Lemma IV.6. *An internal event $p \in \mathbb{C}_A \circledcirc S$ is essential if and only if $s = \Pi_1 p \in S \parallel A$ belongs to S and is internal essential, and $\Pi_2[p] = \widehat{[s]}$.*

Proof. The "if" direction of the proof, is straightforward. For "only if", assume p is essential and internal. By Lemma IV.5, p must project to an internal essential event of either Ⅽ_A or σ. As we have seen, Ⅽ_A has no essential event so it must be that $s = \Pi_1 p \in S$ and is internal essential. Because Ⅽ_A only adds causal links in between the two A component, it follows that Π_1 preserves the causal order between events mapped to S. As a result $\Pi_1[p] = [s] \parallel x_A \in \mathscr{C}(S \parallel A)$. By courtesy of σ, it follows that the maximal visible events of $[s]$ are negative. This implies that $\Pi_2[p] = \widehat{[x]}$ as desired. □

C. Essential strategies

We can now prove that our definition also satisfies *(2)*: all the events added by composition with copycat are inessential:

Theorem IV.7. *Let $\sigma : S \to A \parallel N$ be a uncovered strategy. Then $\text{Ⅽ}_A \circledcirc \sigma \cong \mathscr{E}(\sigma)$.*

Proof sketch. We know that $\sigma_\downarrow$ is a covered strategy so as proved in [14], there is an isomorphism $\varphi : \text{Ⅽ}_A \odot \sigma_\downarrow \cong \sigma_\downarrow$. Lemma III.10 entails that $\text{Ⅽ}_A \odot \sigma_\downarrow \cong (\text{Ⅽ}_A \circledcirc \sigma)_\downarrow$. The difficult part becomes extending φ to the neutral events. By Lemma IV.6, we can let $\varphi(p) = \Pi_1 p$ for an internal essential p. The inverse maps an internal essential $s \in S$ to the prime interaction state $([s] \parallel \sigma_\downarrow[s], \widehat{[s]})$. □

This prompts the following definition:

Definition IV.8. *An uncovered strategy σ is* ***essential*** *if, equivalently:* (1) *all its events are essential,* (2) $\sigma \cong \mathscr{E}(\sigma)$.

It turns out that we can go further and generalize the characterisation of strategies of [14]:

Theorem IV.9. *Let $\sigma : S \to A \parallel N$ be uncovered pre-strategy. It is an essential strategy if and only if $\text{Ⅽ}_A \circledcirc \sigma \cong \sigma$.*

Proof. only if. Obvious from Theorem IV.7.

if. We observe that essential strategies are stable under isomorphism. But $\text{Ⅽ}_A \circledcirc \sigma$ is essential: from Theorem IV.7 and associativity of $\circledcirc$ this boils down to the idempotence of Ⅽ_A for $\circledcirc$, which follows from that for $\odot$ and the fact that by Lemma IV.6, there is no internal essential event in $\mathbb{C}_A \circledcirc \mathbb{C}_A$. □

As a result, we get:

Theorem IV.10. *The following data defines a compact-closed category $\mathbf{CG}_\circledcirc$:*

- objects: *race-free games*
- morphisms from A to B: *essential strategies from A to B (that is essential strategies on $A^\perp \parallel B$) up to isomorphism*

As before, the tensor product is given by parallel composition and duality by duality of games.

1) Relationship between $\mathbf{CG}$ *and* $\mathbf{CG}_{\circledcirc}$*:* Covered strategies can be made into a compact-closed category [14], [4]. Remember that the composition of $\sigma : S \to A^{\perp} \parallel B$ and $\tau : T \to B^{\perp} \parallel C$ in $\mathbf{CG}$ is defined as $\tau \odot \sigma = (\tau \circledast \sigma)_{\downarrow}$.

Lemma IV.11. *The operation* $\sigma \mapsto \sigma_{\downarrow}$ *extends to an identity-on-object functor* $\mathbf{CG}_{\circledcirc} \to \mathbf{CG}$*.*

In the other direction, a strategy $\sigma : A$ might not be an essential strategy – in fact it might not even be an uncovered strategy, as it may fail secrecy. Sending σ to $\text{\ae}_A \circledcirc \sigma$ delegates the non-deterministic choices to internal events and yields an essential strategy, but this operation is not functorial.

2) Relationship between $\mathbf{CG}_{\circledcirc}$ *and* $\mathbf{CG}_{\circledast}$*:* The forgetful operation mapping an essential strategy σ to itself, seen as an uncovered strategy defines a functor $\mathbf{CG}_{\circledcirc} \to \mathbf{CG}_{\circledast}$. Indeed, if two essential strategies are isomorphic, they are also weakly bisimilar. Moreover, we have that $\tau \circledast \sigma \simeq \mathscr{E}(\tau \circledast \sigma) = \tau \circledcirc \sigma$. However the operation $\mathscr{E}(\cdot)$ does not extend to a functor in the other direction even though $\mathscr{E}(\tau) \circledcirc \mathscr{E}(\sigma) \cong \mathscr{E}(\tau \circledast \sigma)$, as it is defined only on concrete representatives, not on equivalence classes for weak bisimilarity.

D. Interpretation of affine IPA

We now show that this new category also supports a sound and adequate interpretation of $\mathbf{aIPA}$. As before, we need to construct the category of negative games and strategies.

Lemma IV.12. *There is a cartesian symmetric monoidal category* $\mathbf{CG}^{-}_{\circledcirc}$ *of negative race-free games and negative essential strategies up to isomorphism. Well-opened negative race-free games form an exponential ideal of* $\mathbf{CG}^{-}_{\circledcirc}$*.*

As a result, we can keep the same interpretation of types of affine IPA. Moreover, all strategies but $[\![\bot]\!]_{\circledast}$ given in Figure 2 are essential. So we keep the same definition for the interpretation, except for $[\![\bot]\!]_{\circledcirc}$ which is the minimal strategy on $[\![\mathbb{B}]\!]$ that contains only the game's minimal negative events:

$$\begin{aligned}
[\![\bot]\!]_{\circledcirc} &= q : \mathbb{B} \\
[\![M; N]\!]_{\circledcirc} &= \mathtt{seq} \circledcirc ([\![M]\!]_{\circledcirc} \parallel [\![N]\!]_{\circledcirc}) \\
[\![M := \mathtt{tt}]\!]_{\circledcirc} &= \mathtt{write} \circledcirc [\![M]\!]_{\circledcirc} \\
[\![M \parallel N]\!]_{\circledcirc} &= \mathtt{join} \circledcirc ([\![M]\!]_{\circledcirc} \parallel [\![N]\!]_{\circledcirc}) \\
[\![!M]\!]_{\circledcirc} &= \mathtt{read} \circledcirc [\![M]\!]_{\circledcirc} \\
[\![\mathtt{if}\ M\, N\, N']\!]_{\circledcirc} &= \mathtt{if} \circledcirc ([\![M]\!]_{\circledcirc} \parallel \langle [\![N]\!]_{\circledcirc}, [\![N']\!]_{\circledcirc} \rangle) \\
[\![\mathbf{newref}\, r := b\, \mathbf{in} M]\!]_{\circledcirc} &= [\![M]\!]_{\circledcirc} \circledcirc \mathbf{cell}_b
\end{aligned}$$

Lemma IV.13. *For all terms* M*, we have* $[\![M]\!]_{\circledcirc} = \mathscr{E}([\![M]\!]_{\circledast})$*.*

Proof. By induction using Lemma IV.4. □

Theorem IV.14. *The interpretation* $[\![\cdot]\!]$ *is sound and adequate for may and must, ie. for any* $\vdash M : \mathbb{X}$*:*

- *the term may converge iff* $[\![M]\!]_{\circledcirc}$ *contains a positive move.*
- *the term must converge iff* $[\![M]\!]_{\circledcirc}$ *must converge.*

Proof. From Lemma IV.13 and IV.2, $[\![M]\!]_{\circledcirc}$ and $[\![M]\!]_{\circledast}$ are bisimilar. The results follows then from the adequacy of $[\![\cdot]\!]_{\circledast}$ (Theorem III.24) and the fact that may and must equivalence are preserved by bisimulation (Lemma III.21). □

V. Conclusion

We described an extension of [14] to uncovered strategies that are composed without hiding. This allows us to have a model with a strong operational flavour: interpreting a language in it is similar to giving it an operational semantics. We have then seen how to extract, from this very operational model, a representation up to weak bisimulation that erases enough information for isomorphism to be meaningful on it.

This mixes well with the work (extension with symmetry, further conditions on strategies) presented in [5], which allows us to generalize the results of [5] to the nondeterministic case: pure computation, even nondeterministic, cannot differentiate a parallel and a sequential implementation of `if` up to may and must convergence. This work, already developed, will appear in the first author's forthcoming PhD thesis.

Acknowledgments

We gratefully acknowledge the support of the ERC Advanced Grant ECSYM.

References

[1] Samson Abramsky, Radha Jagadeesan, and Pasquale Malacaria. Full abstraction for PCF. *Inf. Comput.*, 163(2):409–470, 2000.

[2] Samson Abramsky and Guy McCusker. Linearity, sharing and state: a fully abstract game semantics for Idealized Algol with active expressions. *Electr. Notes Theor. Comput. Sci.*, 3:2–14, 1996.

[3] Simon Castellan and Pierre Clairambault. Causality vs interleaving in game semantics. In *CONCUR 2016 - Concurrency Theory*, 2016.

[4] Simon Castellan, Pierre Clairambault, Silvain Rideau, and Glynn Winskel. Games and strategies as event structures. 2016. Submitted, https://hal.inria.fr/hal-01302713.

[5] Simon Castellan, Pierre Clairambault, and Glynn Winskel. The parallel intensionally fully abstract games model of PCF. In *LICS 2015*. IEEE Computer Society, 2015.

[6] Simon Castellan, Pierre Clairambault, and Glynn Winskel. Concurrent Hyland-Ong games. 2016. https://arxiv.org/abs/1409.7542.

[7] Simon Castellan, Jonathan Hayman, Marc Lasson, and Glynn Winskel. Strategies as concurrent processes. *Electr. Notes Theor. Comput. Sci.*, 308:87–107, 2014.

[8] Clovis Eberhart, Tom Hirschowitz, and Thomas Seiller. Fully-abstract concurrent games for pi. *CoRR*, abs/1310.4306, 2013.

[9] Dan R. Ghica and Andrzej S. Murawski. Angelic semantics of fine-grained concurrency. *Ann. Pure Appl. Logic*, 151(2-3):89–114, 2008.

[10] Russ Harmer. *Games and full abstraction for nondeterministic languages*. PhD thesis, University of London, 1999.

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

[12] James Laird. Full abstraction for functional languages with control. In *LICS*, pages 58–67. IEEE Computer Society, 1997.

[13] James Laird. A game semantics of idealized CSP. *Electr. Notes Theor. Comput. Sci.*, 45:232–257, 2001.

[14] Silvain Rideau and Glynn Winskel. Concurrent strategies. In *LICS*, pages 409–418. IEEE Computer Society, 2011.

[15] Glynn Winskel. Event structures. In Wilfried Brauer, Wolfgang Reisig, and Grzegorz Rozenberg, editors, *Advances in Petri Nets*, volume 255 of *LNCS*, pages 325–392. Springer, 1986.