Imperative Programs as Proofs via Game Semantics [1]

Martin Churchill[b], Jim Laird[a], Guy McCusker[a]

[a]*University of Bath*
[b]*Swansea University*

Abstract

Game semantics extends the Curry-Howard isomorphism to a three-way correspondence: proofs, programs, strategies. But the universe of strategies goes beyond intuitionistic logics and lambda calculus, to capture stateful programs. In this paper we describe a logical counterpart to this extension, in which proofs denote such strategies. The system is expressive: it contains all of the connectives of Intuitionistic Linear Logic, and first-order quantification. Use of Laird's *sequoid* operator allows proofs with imperative behaviour to be expressed. Thus, we can embed first-order Intuitionistic Linear Logic into this system, Polarized Linear Logic, and an imperative total programming language.

The proof system has a tight connection with a simple game model, where games are forests of plays. Formulas are modelled as games, and proofs as history-sensitive winning strategies. We provide a strong *full completeness* result with respect to this model: each finitary strategy is the denotation of a unique analytic (cut-free) proof. Infinite strategies correspond to analytic proofs that are infinitely deep. Thus, we can normalise proofs, via the semantics.

Keywords: game semantics, full completeness, history-sensitive strategies, sequentiality
2010 MSC: 68Q55, 03B70, 03F52, 18C50

1. Introduction

The Curry-Howard isomorphism between proofs in intuitionistic logics and functional programs is a powerful theoretical and practical principle for specifying and reasoning about programs. Game semantics provides a third axis to this correspondence: each proof/program at a given type denotes a strategy for the associated game, and typically a *full completeness* result establishes that this correspondence is also an isomorphism [3]. However, in languages with side-effects such as mutable state it is evident that there are many programs which do not correspond to intuitionistic proofs. Game semantics has achieved notable success in providing models of such programs [5, 2, 20], in which they typically denote "history-sensitive" strategies — strategies which may break the

---

constraints of innocence [14] or history-freeness [3] imposed in fully complete models of intuitionistic or linear logic. The full completeness of these models means there is a precise correspondence between programs and history-sensitive strategies, which raises the question: is there a logic to flesh out the proofs/imperative programs/history-sensitive strategies correspondence?

In this paper we present a first-order logic, WS1, and a games model for it in which proofs denote history-sensitive strategies. Thus total imperative programs correspond, via the game semantics, to proofs in WS1. Moreover, because WS1 is more expressive than the typing system for a typical programming language, it can express finer behavioural properties of strategies. In particular, we can embed first-order intuitionistic logic with equality, Polarized Linear Logic, and a finitary imperative language with ground store, coroutines and some infinite data structures. We also take first steps towards answering some of the questions posed by the logic and its semantics: Are there any formulas which only have 'imperative proofs', but no proofs in a traditional 'functional' proof system? Can we use the expressivity of WS1 to specify imperative programs?

1.1. Related Work

The games interpretation of linear logic upon which WS1 is based was introduced by Blass in a seminal paper [7]. Blass also gives instances of history sensitive strategies which are not denotations of linear logic proofs; these do, however, correspond to proofs in WS1. The particular symmetric monoidal closed category of games underlying our semantics has been studied extensively from both logical and programming perspectives [11, 26, 15]. Longley's project to develop a programming language based on it [30] may be seen as complementary to our aim of understanding it from a logical perspective.

Several logical systems have taken games or interaction as a semantic basis yielding a richer notion of meaning than classical or intuitionistic truth, including Ludics [12] and Computability Logic [18]. The latter also provides an analysis of Blass's examples, suggesting further connections with our logic, although there is a difference of emphasis: the research described here is focused on investigating the structural properties of the games model on which it is based.

Perhaps closest in spirit to our work is tensorial logic, introduced in [34]. Like WS1, tensorial logic is directly inspired by the structure of strategies in game semantics, and in [33], Melliès demonstrates a tight correspondence between the logic and categories of innocent strategies on dialogue games. Our focus in this paper is somewhat different, because we are primarily concerned with the history-sensitive behaviour characteristic of (game semantics of) imperative programs, rather than the purely functional programs that denote innocent strategies.

In [9] a proof theory for Conway games is presented, where formulas are the game trees themselves. In [13], the $\lambda\bar{\lambda}$-calculus is presented, where individual moves of game semantics are represented by variables and binders. Both settings deal with history-sensitive strategies, and have dynamics corresponding to composition of strategies.

A quite different formalisation of game semantics for first order logic is given in [29], also with a full completeness result.

1.2. Contribution

The main contribution of this paper is to present an expressive logical system and its semantics, in which proofs correspond to history sensitive strategies. Illustrating the

2

expressive power of this system, we show how proofs of intuitionistic first-order logic, Polarized Linear Logic and imperative programming constructs may be embedded in it. We also demonstrate how formulas in the logic can be used to represent some properties of imperative programs: for example, we describe a formula for which any proof corresponds to a well-behaved (single write) Boolean storage cell.

The interpretation of WS1 includes some interesting developments of game semantics. In particular, the exponentials are treated in a novel way: we use the fact that the semantic exponential introduced in [15] is a final coalgebra, and reflect this explicitly in the logic in the style of [8]. This formulation allows us to express the usual exponential introduction rules (promotion and dereliction) but also proofs that correspond to strategies on $!A$ that act differently on each interrogation, such as the reusable Boolean reference cell. Another development is the interpretation of first-order logic with equality. A proof corresponds to a family of winning strategies — one for each possible interpretation of the atoms determined by a standard notion of $\mathcal{L}$-structure — which must be *uniform* across $\mathcal{L}$-structures. This notion of uniformity is precisely captured by the requirement that strategies are *lax natural transformations* between the relevant functors.

The main technical results of this paper concern the sharp correspondence between proofs and strategies: *full completeness* results. We show that any bounded uniform winning strategy is the denotation of a unique (cut-free) *analytic proof.* In the exponential-free fragment, where all strategies are bounded, it follows that many rules such as cut are admissible; and it allows us to normalise proofs to analytic proofs via the semantics. For the full logic, since the exponentials correspond to final coalgebras, proofs can be unfolded to infinitary form. Extending semantics-based normalisation to the full WS1, the resulting normal forms are *infinitary* analytic proofs.

2. Games and Strategies

Our notion of game is essentially that introduced by [7], and similar to that of [3, 25], augmented with winning conditions introduced as in [15]. We make use of the categorical structure on games and strategies first introduced in [19].

Informally, a game is a tree where Player and Opponent own alternate nodes, together with a polarity specifying which protagonist owns the starting node. A play proceeds down a particular branch, with Opponent/Player choosing the subtree for nodes they control. A strategy for Player specifies which choice Player should make in response to Opponent's moves so far. The winner of a finite play is the last protagonist to play a move. The winner of an infinite play is specified by a winning condition for each game.

If A is a set, let A^* denote the free monoid (set of sequences) over A, A^ω the set of infinite sequences over A, and ϵ the empty sequence. We write $s \sqsubseteq t$ if s is a prefix of s, and $s \sqsubset t$ if s is a strict (finite) prefix of (possibly infinite) t. If $X \subseteq A^*$, write $\overline{X} = \{s \in A^\omega : \forall t \sqsubset s, t \in X\}$.

Definition A *game* is a tuple $(M_A, \lambda_A, b_A, P_A, W_A)$ where

- M_A is a set of moves

- $\lambda_A : M_A \to \{O, P\}$

  – We call m an *O-move* if $\lambda_A(m) = O$ and a *P-move* if $\lambda_A(m) = P$.

3

- $b_A \in \{O, P\}$ specifies a starting player

  - We call $s \in M_A^*$ *alternating* if s starts with a b_A-move and alternates between O-moves and P-moves. Write $M_A^{\circledast}$ for the set of such sequences.

- $P_A \subseteq M_A^{\circledast}$ is a nonempty prefix-closed set of valid plays.

- $W_A \subseteq \overline{P_A}$ represents the set of infinite plays that are P-winning; we say an infinite play is O-winning if it is not P-winning.

For finite plays, the last player to play a move wins: let $W_A^* = W_A \cup E_A$ where E_A is the set of plays that end in a P-move. We will call a game A *negative* if $b_A = O$ and *positive* if $b_A = P$. We write $A, B, C, \ldots$ for arbitrary games; $L, M, N, \ldots$ for arbitrary negative games and $P, Q, R, \ldots$ for arbitrary positive games.

Definition If A is a game, we define its *negation* by changing its polarity, and swapping its Player/Opponent labelling. Define $\neg : \{O, P\} \to \{O, P\}$ by $\neg(O) = P$ and $\neg(P) = O$.

$$A^{\perp} = (M_A, \neg \circ \lambda_A, \neg b_A, P_A, \overline{P_A} - W_A).$$

Negation is evidently an involutive bijection between negative and positive games.

Definition A *strategy* σ for a game $(M_A, \lambda_A, b_A, P_A, W_A)$ is a subset of P_A (a set of traces) satisfying:

- If $sa \in \sigma$, then $\lambda_A(a) = P$

- If $sab \in \sigma$, then $s \in \sigma$

- If $sa, sb \in \sigma$, then $a = b$

- If $\sigma = \varnothing$ then $b_A = P$, and if $\epsilon \in \sigma$ then $b_A = O$.

We say a strategy σ is *bounded* if $\exists k \in \mathbb{N}. \forall s \in \sigma. |s| \leq k$; in which case we write $\mathsf{depth}(\sigma)$ for the smallest such k (the length of the longest play in σ).

Definition A strategy on a game A is *total* if it is nonempty and whenever $s \in \sigma$ and $sa \in P_A$, there is some $b \in M_A$ such that $sab \in \sigma$. A total strategy σ is *winning* if whenever $s \in \overline{P_A}$ and all prefixes of s ending in a P-move are in σ, then $s \in W_A$.

2.1. Connectives

We next describe operations on games, which will correspond to connectives in our logic. These come in dual pairs, determined by involutive negation.

First, some notation. If X and Y are sets, let $X + Y = \{\mathsf{in}_1(x) : x \in X\} \cup \{\mathsf{in}_2(y) : y \in Y\}$. We use standard notation $[f, g]$ for copairing. If $s \in (X + Y)^*$ or $s \in (X + Y)^{\omega}$ then $s|_i$ is the subsequence of s consisting of elements of the form $\mathsf{in}_i(z)$. If $X_1 \subseteq X^*$ and $Y_1 \subseteq Y^*$ let $X_1 \| Y_1 = \{s \in (X + Y)^* : s|_1 \in X_1 \wedge s|_2 \in Y_1\}$. If $X_1 \subseteq X^{\omega}$ and $Y_1 \subseteq Y^{\omega}$ let $X_1 \| Y_1 = \{s \in (X + Y)^{\omega} : s|_1 \in X_1 \wedge s|_2 \in Y_1\}$.

Empty Game. We define a negative game with no moves:

$$\mathbf{1} = (\emptyset, \emptyset, O, \{\epsilon\}, \emptyset).$$

There is one strategy on $\mathbf{1}$ given by $\{\epsilon\}$, and this strategy is total (and winning, as $\overline{P_{\mathbf{1}}}$ is empty).

There is one strategy, $\emptyset$, on the empty *positive* game $\mathbf{0} = \mathbf{1}^{\perp}$. This strategy is not total (intuitively, it is Player's turn to play first but he has no moves to play).

One-move Game. We write $\perp$ for the negative game with a single move q and maximal play consisting of q :

$$\perp = (\{q\}, \{q \mapsto O\}, O, \{\epsilon, q\}, \emptyset).$$

There is a single strategy $\{\epsilon\}$ on $\perp$; this is not total.

We write $\top$ for the positive game with a single move, $\perp^{\perp}$. There are two strategies on $\top$: $\varnothing$ (which is evidently not total) and $\{q\}$ which is total (and thus, trivially winning).

Disjoint Union. The negative game $L \& N$ is played over the disjoint union of the moves of L and N: a play in this game is either a (tagged) play in L or a (tagged) play in N. A play is P-winning if it is a P-winning play from L or a P-winning play from N. Thus, on Opponent's first move he chooses to play either in L or N, and thereafter play remains in that component. Formally, define

$$L \& N = (M_L + M_N, [\lambda_L, \lambda_N], O, P_L +^* P_N, \{\mathsf{in}_1^\omega(s) : s \in W_L\} \cup \{\mathsf{in}_2^\omega(s) : s \in W_N\})$$

where $X_1 +^* Y_1 = \{s \in X_1 \| Y_1 : s|_1 = \epsilon \vee s|_2 = \epsilon\}$ if $X_1 \subseteq X^*$ and $Y_1 \subseteq Y^*$, and if $s \in X_i^*$ (resp. X_i^ω) we write $\mathsf{in}_i^*(s)$ (resp. in_i^ω) for the corresponding sequence in $(X_1 + X_2)^*$ (resp. $(X_1 + X_2)^\omega$). A (winning) strategy on $L \& N$ corresponds to a pairing of a (winning) strategy on L with a (winning) strategy on N — hence the identification of this connective with the "with" of linear logic.

Similarly, the positive game $Q \oplus R = (Q^{\perp} \& R^{\perp})^{\perp}$ corresponds to a disjoint union of plays from Q and R where Player's first move constitutes a choice to play either in Q or R. An infinite play in $Q \oplus R$ is P-winning if it is P-winning in the relevant component. Thus a winning strategy on $Q \oplus R$ corresponds to either a winning strategy on Q or a winning strategy on R.

We may form any set-indexed conjunctions and disjunctions in this way. Let X be a set and $\{N_x : x \in X\}$ a family of negative games indexed by X. We define the game $\prod_{x \in X} N_x$ by

$$(\sum_{x \in X} M_{N_x}, \mathsf{in}_x(m) \mapsto \lambda_{N_x}(m), O, \{\mathsf{in}_x^*(s) : x \in X, s \in P_{N_x}\}, \{\mathsf{in}_x^\omega(s) : x \in X, s \in W_{N_x}\}).$$

If $\{Q_x : x \in X\}$ is a family of positive games then $\bigoplus_{x \in X} Q_x = (\prod_{x \in X} N_x^{\perp})^{\perp}$.

Symmetric Merge. If L and N are negative games, a play in the negative game $L \otimes N$ is an *interleaving* of a play in L with a play in N. Define

$$L \otimes N = (M_L + M_N, [\lambda_L, \lambda_N], O, (P_L \| P_N) \cap M_{L \otimes N}^{\circledast}, \{s \in \overline{P_{L \otimes N}} : s|_1 \in W_L^* \wedge s|_2 \in W_N^*\}).$$

The fact that the play restricted to each component must be alternating, and that the play overall must be alternating, ensures that only Opponent may switch between

5

components. This operation may be used to interpret the "times" of linear logic [7]. An infinite play in $L \otimes N$ is P-winning if its restriction to L is P-winning and its restriction to N is P-winning.

Similarly, if Q and R are positive games, plays in the positive game $Q \parr R = (Q^\perp \otimes R^\perp)^\perp$ consist of interleavings of plays in Q and R in which Player may switch between the two components. An infinite play in $Q \parr R$ is P-winning if its restriction to Q is P-winning or its restriction to R is P-winning.

Left Merge. Let A be a game of polarity a (positive or negative), and N a negative game. The game $A \oslash N$ has polarity a: a play in this game is an interleaving of a play in A with a play in N *such that the first move, if any, is in A*. An infinite play in $A \oslash N$ is P-winning if both of its restrictions are P-winning. Formally, define

$$A \oslash N = (M_A + M_N, [\lambda_A, \lambda_N], b_A, (P_A \|_L P_N) \cap M_{A \oslash N}^\circledast, \{s \in P_{A \oslash N}^\omega : s|_1 \in W_A^* \wedge s|_2 \in W_N^*\}).$$

where $X_1 \|_L Y_1 = \{s \in X_1 \| Y_1 : s|_1 = \epsilon \Rightarrow s|_2 = \epsilon\}$. Observe that it is Opponent who switches between components: if A is negative then $A \oslash N$ consists of the plays in $A \otimes N$ which start in A (or are empty). This connective on games, the *sequoid*, was introduced in [21] and its properties can be used to model stateful effects [21, 24].

If Q is a positive game, the game $A \lhd Q = (A^\perp \oslash Q^\perp)^\perp$ has the same polarity as A, and consists of interleavings of a play in A and a play in Q, starting in A and with Player switching between components and winning an infinite play if he wins in either A or Q.

Exponentials. Let N be a negative game. The negative game $!\,N$ consists of countably many copies of N, tagged with natural numbers. A play over $!\,N$ is an interleaving of plays in each copy, such that any move in N_{i+1} is preceded by a move in N_i. An infinite play is winning just if it is winning in each component. Define

$$!\,N = (M_N \times \mathbb{N}, \lambda_N \circ \pi_1, \{s : \forall i.s|_i \in P_N \wedge s|_i = \epsilon \Rightarrow s|_{i+1} = \epsilon\}, \{s : \forall i.s|_i \in W_N^*\}).$$

As with the tensor, there is an implicit switching condition: only Opponent can open new copies and switch between copies. This operation may be used to interpret the "of course" of linear logic [15].

Dually, if Q is a positive game, $?\,Q = (!\,Q^\perp)^\perp$ is the game consisting of an infinite number of copies of Q, where Player can spawn new copies and switch between them. An infinite play in $?\,Q$ is winning if it is winning in at least one component.

2.1.1. Derived Connectives

We shall also make use of the following derived operations:

Lifts. We can use left merge to add a single move at the beginning of a game. If N is a negative game, a play in the positive game

$$\downarrow N = \top \oslash N$$

consists of a play in N prefixed by an extra P-move. A strategy on $\downarrow N$ is either $\emptyset$ or corresponds to a strategy on N. A winning strategy on $\downarrow N$ corresponds to a winning strategy on N. If P is a positive game, a play in the negative game

$$\uparrow P = \perp \lhd P$$

consists of a play in P prefixed by an extra O-move. A (winning) strategy on $\uparrow P$ corresponds to a (winning) strategy on P.

Affine Implication. If M and N are negative games, we may define

$$M \multimap N = N \lhd M^{\perp}.$$

A play in $M \multimap N$ consists of a play in N interleaved with a play in $M^{\perp}$ (an 'input version' of M), starting in N. It is winning if its restriction to N is P-winning or its restriction to $M^{\perp}$ is P-winning (i.e. its restriction to M is O-winning), agreeing with [15].

2.1.2. Isomorphisms of Games

Given two games A and B, we say that A and B are *forest isomorphic* if $b_A = b_B$ and there is a bijection from P_A to P_B which is monotone with respect to the prefix order, and restricts to a bijection on the P-winning plays. Some forest isomorphisms between games are given in Figure 1. Each isomorphism $M \cong N$ gives rise to winning strategies $M \multimap N$ and $N \multimap M$, which are mutually inverse. Thus, winning strategies on M are in bijective correspondence with winning strategies on N.

Figure 1: Some Characteristic Isomorphisms of Games

$M \otimes N \cong N \otimes M$	$P \parr Q \cong Q \parr P$
$M \otimes (N \otimes L) \cong (M \otimes N) \otimes L$	$P \parr (Q \parr R) \cong (P \parr Q) \parr L$
$M \otimes \mathbf{1} \cong M \cong M \& \mathbf{1}$	$P \parr \mathbf{0} \cong P \cong P \oplus \mathbf{0}$
$M \& N \cong N \& M$	$P \oplus Q \cong Q \oplus P$
$M \& (N \& L) \cong (M \& N) \& L$	$P \oplus (Q \oplus R) \cong (P \oplus Q) \oplus R$
$(M \otimes N) \multimap L \cong M \multimap (N \multimap L)$	$P \oslash (M \otimes N) \cong (P \oslash M) \oslash N$
$M \multimap (N \& L) \cong (M \multimap N) \& (M \multimap L)$	$(P \oplus Q) \oslash N \cong (P \oslash N) \oplus (Q \oslash N)$
$M \multimap \mathbf{1} \cong \mathbf{1} \oslash M \cong \mathbf{1}$	$\mathbf{0} \oslash M \cong \mathbf{0} \lhd M^{\perp} \cong \mathbf{0}$
$M \otimes N \cong (M \oslash N) \& (N \oslash M)$	$P \parr Q \cong (P \lhd Q) \oplus (Q \lhd P)$
$(M \& N) \oslash L \cong (M \oslash L) \& (N \oslash L)$	$(P \oplus Q) \lhd R \cong (P \lhd R) \oplus (Q \lhd R)$
$\prod_{x \in X} (M_x \oslash L) \cong \prod_{x \in X} M_x \oslash L$	$\bigoplus_{x \in X} (P_x \lhd R) \cong \bigoplus_{x \in X} P_x \lhd R$
$M \oslash (N \otimes L) \cong (M \oslash N) \oslash L$	$P \lhd (Q \parr R) \cong (P \lhd Q) \lhd R$
$M \oslash \mathbf{1} \cong M$	$P \lhd \mathbf{0} \cong P$
$(M \multimap N) \multimap \perp \cong (N \multimap \perp) \oslash M$	$\top \oslash (M \lhd Q) \cong (\top \oslash M) \lhd Q$
$\perp \oslash M \cong \perp$	$\top \lhd P \cong \top$
$!N \cong N \oslash !N$	$?P \cong P \lhd ?P$
$!(N \& M) \cong !N \otimes !M$	$?(P \oplus Q) \cong ?P \parr ?Q$

2.2. Imperative Objects as Strategies

We may model higher-order programming languages with imperative features by interpreting *types* as games and *programs* as strategies. (Such a semantics of a full object-oriented language, using essentially the notion of game described here, is described in [38].) Here, we illustrate the capacity of our games and strategies to represent imperative objects by describing a strategy with the behaviour of a Boolean reference cell, on a game

corresponding to the type of imperative Boolean variables — essentially the *cell* strategy first described, for a different notion of game, in [5]. (We will later see how this strategy can be represented as a proof in our logic.)

Let $\mathbf{B} = \bot \lhd \top \oplus \top$ be the (negative) game of "Boolean output" — this has one initial Opponent-move q and two possible Player responses, representing `True` or `False`. Let $\mathbf{Bi} = (\bot \& \bot) \lhd \top$ be the (negative) game of "Boolean input" which has two starting Opponent-moves `in(tt)` and `in(ff)` and one possible response to this, `ok`. The game $!(\mathbf{Bi}\&\mathbf{B})$ represents the type of a Boolean variable — it is a product of a `write` method which accepts a Boolean input and a `read` method which on interrogation produces a Boolean output, under an exponential which allows these methods to be used arbitrarily many times.

The strategy *cell* on this game represents a reference cell which accepts Boolean input on the left, and returns the last value written to it as output on the right (we assume it is initialised with `ff`). For readability, we will omit the tags on the product and the exponential (since they can be inferred).

$$!(\mathbf{Bi} \quad \& \quad \mathbf{B})$$

	q	O
	ff	P
in(tt)		O
ok		P
	q	O
	tt	P

In contrast with the *history-free* strategies which denote proofs of linear logic in the model of [3], this strategy is *history-sensitive* — the move prescribed by the strategy depends on the entire play so far. It is this property which allows the state of the object to be described implicitly, as in [5].

3. The Logic WS1

3.1. Formulas of WS1

The formulas of WS1 are based on first-order linear logic, with some additional connectives, and subject to a notion of polarity. A *first-order language* consists of:

- A collection of complementary pairs of predicate symbols ϕ (negative) and $\overline{\phi}$ (positive), each with an arity in $\mathbb{N}$ such that $\mathsf{ar}(\phi) = \mathsf{ar}(\overline{\phi})$. This must include the binary symbol $=$ (negative), and we write $\neq$ for its complement

- A collection of function symbols, each with an arity.

The negative and positive formulas of WS1 over $\mathcal{L}$ are defined by the following grammar. M, N range over negative formulas and P, Q over positive formulas; variables range over some global set $\mathcal{V}$.

$$
\begin{array}{llllll}
M, N := & \mathbf{1} & | & \bot & | & \phi(\overrightarrow{s}) & | \\
& M \otimes N & | & M \oslash N & | & N \lhd P & | \\
& \forall x.N & | & M \& N & | & !N & \\
P, Q := & \mathbf{0} & | & \top & | & \overline{\phi}(\overrightarrow{s}) & | \\
& P \parr Q & | & P \lhd Q & | & P \oslash N & | \\
& \exists x.P & | & P \oplus Q & | & ?P &
\end{array}
$$

Here, s ranges over $\mathcal{L}$-terms, x over variables, and $\phi(\overrightarrow{s})$ over n-ary predicates ϕ applied to a tuple of terms $\overrightarrow{s} = (s_1, \ldots, s_n)$.

The involutive negation operation $(\_)^{\perp}$ sends negative formulas to positive ones and *vice versa* by exchanging each atom, unit or connective for its dual — i.e. $\mathbf{1}$ for $\mathbf{0}$, $\bot$ for $\top$, $\phi(\overrightarrow{x})$ for $\overline{\phi}(\overrightarrow{x})$, $\otimes$ for $\parr$, $\oslash$ for $\lhd$, $\forall$ for $\exists$, $\&$ for $\oplus$ and $!$ for $?$.

3.1.1. Interpreting Formulas as Games

We may interpret each positive formula as a positive game, and each negative formula as a negative game, by fixing a truth assignment for the atomic formulas via a standard notion of first-order structure.

Definition An $\mathcal{L}$-structure L is a set $|L|$ together with an interpretation function I_L sending:

- each predicate symbol (with arity n) to a function $|L|^n \to \{\mathtt{tt}, \mathtt{ff}\}$ such that $I_L(\phi)(\overrightarrow{a}) \neq I_L(\overline{\phi})(\overrightarrow{a})$ for all $\vec{a}$ and $I_L(=)(a, b) = \mathtt{tt}$ iff $a = b$;

- each function symbol f (with arity n) to a function $I_L(f) : |L|^n \to |L|$.

For any $X \subseteq \mathcal{V}$, an $\mathcal{L}$-*model over* X is a pair (L, v) where L is an $\mathcal{L}$-structure and $v : X \to |L|$ a valuation function, yielding an assignment of truth values to all atomic formulas with variables in X.

Given a $\mathcal{L}$-model (L, v) over X, we may interpret each formula A with free variables in X as a game $[\![A]\!](L, v)$ in as follows:

- Each of the units and connectives $\otimes, \parr, \oslash, \lhd, \mathbf{1}, \mathbf{0}, \top, \bot, !, ?, \&, \oplus$ is interpreted as the corresponding operation on games from Section 2.1, lifted to an action on families of games.

- Positive atoms which are assigned *true* in (L, v) are interpreted as the game with a single (Player) move ($\top$); positive atoms which are assigned *false* are interpreted as the game with no moves ($\mathbf{0}$). Conversely, negative atoms which are assigned *true* in (L, v) are interpreted as the empty game ($\mathbf{1}$), whilst negative atoms which are assigned *false* are interpreted as the game with a single Opponent move ($\bot$).

- Quantifiers are interpreted as additive conjunctions and disjunctions over the domain of L — i.e. $[\![\forall x.N]\!](L, v) = \prod_{l \in |L|} [\![N]\!](L, v[x \mapsto l])$ and $[\![\exists x.P]\!](L, v) = \bigoplus_{l \in |L|} [\![P]\!](L, v[x \mapsto l])$. In the case of $\forall x.N$, this is equivalent to Opponent choosing an $x \in |L|$ and play proceeding in $N(x)$. In the case of $\exists x.P$, this is equivalent to Player choosing an $x \in |L|$ and play proceeding in $P(x)$.

Note that $[\![A^{\perp}]\!] = [\![A]\!]^{\perp}$.

3.2. Proofs

A proof of a formula $\vdash A$ will be interpreted as a *uniform family* of *winning strategies* on $\llbracket A \rrbracket(L, v)$ for each (L, v). We will formalise this interpretation (and, importantly, the meaning of "uniformity") in Section 6, but with this in mind, we can define proof rules for WS1. A *sequent* of WS1 is of the form $X; \Theta \vdash \Gamma$ where $X \subseteq \mathcal{V}$, Θ is a set of positive atomic formulas and Γ is a nonempty list of formulas such that $FV(\Theta, \Gamma) \subseteq X$. The explicit free variable set X is required for the tight correspondence between the syntax and semantics. For brevity, let Φ range over $X; \Theta$ contexts.

We shall interpret such a sequent as a (family of) dialogue games by interpreting the comma operator in Γ as left-associative left-merge (i.e. either $\oslash$ or $\lhd$ depending on the polarity of the right-hand operand), so that the first move must occur in the first element (or head formula) of Γ. For example, if M, N are negative formulas and P, Q positive formulas, the sequent

$$\vdash M, P, Q, N$$

is semantically equivalent to

$$\vdash ((M \lhd P) \lhd Q) \oslash N.$$

Thus, in the game interpretation of a sequent Γ the first move must occur in the first (or *head*) formula of Γ.

The derivation rules for proofs are partitioned into *core rules* and *other rules*. Here M, N range over negative formulas, P, Q over positive formulas, Γ, Δ over lists of formulas, Γ^* over non-empty lists of formulas and Γ^+, Δ^+ over lists of positive formulas.

3.2.1. Core Rules

Each n-ary connective $\odot$ of WS1 is associated with *core introduction rules* which introduce that connective in the head position of a sequent: they conclude $\Phi \vdash \odot(A_1, \ldots, A_n), \Gamma$ from some premises. These rules are given in Figure 2. These core introduction rules are all *additive* (by contrast to linear logic: note in particular the difference with respect to the $\otimes$ introduction rule).

We may interpret each of the core introduction rules with respect to (L, v) as follows:

- The interpretation of $\mathsf{P_1}$ is the unique total strategy on the game $\mathbf{1}, \Gamma$ (where it is Opponent's turn to start, but there are no moves for him to play since the first move must take place in the empty game $\mathbf{1}$).

- The interpretation of $\mathsf{P_\top}$ is the unique total strategy on the game $\top$, where Player plays a move and the game is over.

- The interpretation of the unary rule $\mathsf{P_\oslash}$ is the identity function, as the game denoted by the conclusion is the same game as that denoted by the premise. The interpretation of $\mathsf{P_\lhd}$ is similar.

- For $\mathsf{P_\&}$ we note that given strategies $\sigma : M, \Gamma$ and $\tau : N, \Gamma$ we can construct a strategy on $M \& N, \Gamma$ which plays as σ if Opponent's first move is in M, and as τ if Opponent's first move is in N.

Figure 2: Core Introduction Rules for WS1

$$\mathsf{P_1} \ \frac{}{\Phi \vdash \mathbf{1}, \Gamma} \qquad\qquad \mathsf{P_\oslash} \ \frac{\Phi \vdash A, N, \Gamma}{\Phi \vdash A \oslash N, \Gamma} \qquad\qquad \mathsf{P_\lhd} \ \frac{\Phi \vdash A, P, \Gamma}{\Phi \vdash A \lhd P, \Gamma}$$

$$\mathsf{P_\otimes} \ \frac{\Phi \vdash M, N, \Gamma \qquad \Phi \vdash N, M, \Gamma}{\Phi \vdash M \otimes N, \Gamma} \qquad \mathsf{P_{\otimes 1}} \ \frac{\Phi \vdash P, Q, \Gamma}{\Phi \vdash P \otimes Q, \Gamma} \qquad \mathsf{P_{\otimes 2}} \ \frac{\Phi \vdash Q, P, \Gamma}{\Phi \vdash P \otimes Q, \Gamma}$$

$$\mathsf{P_\&} \ \frac{\Phi \vdash M, \Gamma \qquad \Phi \vdash N, \Gamma}{\Phi \vdash M \& N, \Gamma} \qquad\quad \mathsf{P_{\oplus 1}} \ \frac{\Phi \vdash P, \Gamma}{\Phi \vdash P \oplus Q, \Gamma} \qquad\quad \mathsf{P_{\oplus 2}} \ \frac{\Phi \vdash Q, \Gamma}{\Phi \vdash P \oplus Q, \Gamma}$$

$$\mathsf{P_\top} \ \frac{}{\Phi \vdash \top} \qquad\qquad\quad \mathsf{P_\top^-} \ \frac{\Phi \vdash N}{\Phi \vdash \top, N} \qquad\qquad\quad \mathsf{P_\bot^+} \ \frac{\Phi \vdash P}{\Phi \vdash \bot, P}$$

$$\mathsf{P_{at-}} \ \frac{X; \Theta, \overline{\phi}(\overrightarrow{s}) \vdash \bot, \Gamma}{X; \Theta \vdash \phi(\overrightarrow{s}), \Gamma} \qquad \mathsf{P_{at+}} \ \frac{X; \Theta, \overline{\phi}(\overrightarrow{s}) \vdash \top, \Gamma}{X; \Theta, \overline{\phi}(\overrightarrow{s}) \vdash \overline{\phi}(\overrightarrow{s}), \Gamma} \qquad \mathsf{P_!} \ \frac{\Phi \vdash N, !N, \Gamma}{\Phi \vdash !N, \Gamma}$$

$$\mathsf{P_\exists^s} \ \frac{X; \Theta \vdash P[s/x], \Gamma}{X; \Theta \vdash \exists x.P, \Gamma} \ FV(s) \subseteq X \qquad \mathsf{P_\forall} \ \frac{X \uplus \{x\}; \Theta \vdash N, \Gamma}{X; \Theta \vdash \forall x.N, \Gamma} \qquad \mathsf{P_?} \ \frac{\Phi \vdash P, ?P, \Gamma}{\Phi \vdash ?P, \Gamma}$$

- Similarly, for $\mathsf{P_\otimes}$ we note that given strategies $\sigma : M, N, \Gamma$ and $\tau : N, M, \Gamma$ we can construct a strategy on $M \otimes N$ which plays as σ if Opponent's first move is in M, and as τ if Opponent's first move is in N. Here we are making use of the isomorphism $M \otimes N \cong (M \oslash N) \& (N \oslash M)$ — each play in $M \otimes N$ must either start in M (and thus be a play in $M \oslash N$) or in N (and thus be a play in $N \oslash M$). Thus, WS1 commits to a particular interpretation of $\otimes$, rather than an arbitrary monoidal structure.

- For $\mathsf{P_{\oplus 1}}$ we note that given a strategy $\sigma : P, \Gamma$ we can construct a strategy on $P \oplus Q, \Gamma$ with Player choosing to play his first move in P and thereafter playing as σ. For $\mathsf{P_{\oplus 2}}$ Player can play his first move in Q and then play as the given strategy.

- Similarly, for the $\mathsf{P_\otimes}$ rules, we note that in a strategy on $P \otimes Q, \Gamma$ Player may choose to either play his first move in P (requiring a strategy on P, Q, Γ) or in Q (requiring a strategy on Q, P, Γ).

- The interpretation of $\mathsf{P_\bot^+}$ uses the observation that total strategies on $\bot, P = \uparrow P$ are in correspondence with total strategies on P. Similarly, the interpretation of $\mathsf{P_\top^-}$ uses the observation that total strategies on $\top, N = \downarrow N$ are in correspondence with total strategies on N.

- For $\mathsf{P_{at-}}$, we know that $\phi(\overrightarrow{s}), \Gamma$ is interpreted by $\mathbf{1}, \Gamma$ if $(L, v) \models \phi(\overrightarrow{s})$ and by $\bot, \Gamma$ if $(L, v) \not\models \phi(\overrightarrow{s})$. In the former case, there are no moves to respond to, so we only need to consider the case when $(L, v) \models \overline{\phi}(\overrightarrow{s})$.

- For $\mathsf{P_{at+}}$, we can only provide a family of strategies on a game whose first move is in $\overline{\phi}(\overrightarrow{s})$ if we know that $(L, v) \models \overline{\phi}(\overrightarrow{s})$ since otherwise our family has to contain a winning strategy on the empty positive game $\mathbf{0}$, of which there are none.

- For $\mathsf{P_\forall}$, to give a family of strategies on $\forall x.N, \Gamma$ we must give a strategy on N, Γ for each choice of x — that is, a family of strategies on the set of Θ-satisfying $\mathcal{L}$-models over $X \uplus \{x\}$.

$$\mathsf{P}_\perp^- \; \frac{\Phi \vdash \perp, \Gamma}{\Phi \vdash \perp, N, \Gamma} \qquad \mathsf{P}_\perp^{\lower2pt\hbox{$\scriptstyle\gamma\!\!\!\gamma$}} \; \frac{\Phi \vdash \perp, P \,\lower2pt\hbox{$\scriptstyle\gamma\!\!\!\gamma$}\, Q, \Gamma}{\Phi \vdash \perp, P, Q, \Gamma} \qquad \mathsf{P}_\perp^\oslash \; \frac{\Phi \vdash \perp, P \oslash N, \Gamma}{\Phi \vdash \perp, P, N, \Gamma}$$

$$\mathsf{P}_\top^+ \; \frac{\Phi \vdash \top, \Gamma}{\Phi \vdash \top, P, \Gamma} \qquad \mathsf{P}_\top^\otimes \; \frac{\Phi \vdash \top, M \otimes N, \Gamma}{\Phi \vdash \top, M, N, \Gamma} \qquad \mathsf{P}_\top^\lhd \; \frac{\Phi \vdash \top, N \lhd P, \Gamma}{\Phi \vdash \top, N, P, \Gamma}$$

$$\mathsf{P}_{\mathsf{ma}}^{x,y,z} \; \frac{(X; \Theta \vdash \Gamma)[\tfrac{z}{x}, \tfrac{z}{y}] \qquad X; \Theta, x \neq y \vdash \Gamma}{X; \Theta \vdash \Gamma} \qquad \mathsf{P}_{\neq} \; \frac{}{X; \Theta, x \neq x \vdash \Gamma}$$

- For $\mathsf{P}_\exists$, to give a family of strategies on $\exists x. P, \Gamma$ we must choose a value s for x and give a family of strategies on $P[s/x], \Gamma$.

As well as the core introduction rules, there is a small set of *core elimination rules*, found in Figure 3. These permit decomposition of the second and third formula in a sequent, if the first formula is $\perp$ or $\top$. They correspond to isomorphisms between the premise and conclusion in the semantics, which induces a bijection between the winning strategies on each. For example, $\mathsf{P}_\perp^-$ uses the isomorphism $\perp \oslash N \cong \perp$, and $\mathsf{P}_\perp^{\lower2pt\hbox{$\scriptstyle\gamma\!\!\!\gamma$}}$ the isomorphism $\perp \lhd (P \,\lower2pt\hbox{$\scriptstyle\gamma\!\!\!\gamma$}\, Q) \cong (\perp \lhd P) \lhd Q$ and $\mathsf{P}_\perp^\oslash$ the isomorphism $\perp \lhd (P \oslash N) \cong (\perp \lhd P) \oslash N$.

Finally, there are *core equality rules* which deal with equality, given in Figure 4. We can interpret the core equality rules at a model (L, v) as follows:

- To interpret $\mathsf{P}_{\neq}$ (reflexivity of identity), we take the empty family of strategies, since there are no Θ-satisfying $\mathcal{L}$-models if Θ contains $x \neq x$.

- To interpret the matching rule $\mathsf{P}_{\mathsf{ma}}^{x,y,z}$, we note that the collection of Θ-satisfying $\mathcal{L}$-models can be decomposed into those where x and y are identified (the left-hand premise) and those where they are distinct (the right-hand premise).

Once a discipline regarding where the matching rule is applied has been introduced, proof search in this core subsystem is particularly simple, as the form of the sequent to be proved determines the choice of final rule. We will later show that the core rules are sufficient to denote any finitary family of uniform winning strategies.

We make a brief note on polarities and reversibility, and a comparison with focused proof systems. In such systems, polarisation is used to differentiate between connectives whose corresponding rules are *reversible* or *irreversible* [6]. Irreversible rules act on positive formulas. An irreversible rule is one where (reading upwards) in applying the rule one must make some definite choice, a choice which could determine whether the proof search succeeds or not. Thus, additive disjunction introduction is always an irreversible rule, and in linear logic so is the tensor introduction rule, since a choice must be made regarding how the context is split.

In WS1, the core introduction rule for tensor (as for all such rules) is additive, not multiplicative. Thus, this rule is reversible, and $\otimes$ is resultantly a negative connective. In contrast, $\invamp$ is a positive connective as there are two different core introduction rules, which are not reversible. Thus, as well as the semantic motivation, we can view our distinction between positive and negative formulas in the same light as the polarities of focused systems.

However, there is an important distinction. In focused systems, the proof search alternates between negative phases, in which reversible rules are applied, and positive phases, in which irreversible rules are applied. Analytic proof search in WS follows a different two-phase discipline, in which we first *decompose* the first formula of a sequent into a unit using the core introduction rules, and then *collate* the tail formulas together using the core elimination rules. We will give an embedding of LLP inside WS in Section 3.3.

3.2.2. Other Rules

Figure 5: Non-core rules of WS1

$$\mathsf{P_1^T} \; \frac{\Phi \vdash \Gamma^*, \Delta}{\Phi \vdash \Gamma^*, \mathbf{1}, \Delta} \qquad \mathsf{P_\otimes^T} \; \frac{\Phi \vdash \Gamma^*, M, N, \Delta}{\Phi \vdash \Gamma^*, M \otimes N, \Delta} \qquad \mathsf{P_{sym}^-} \; \frac{\Phi \vdash \Gamma^*, M, N, \Delta}{\Phi \vdash \Gamma^*, N, M, \Delta} \qquad \mathsf{P_{wk}^-} \; \frac{\Phi \vdash \Gamma^*, M, \Delta}{\Phi \vdash \Gamma^*, \Delta}$$

$$\mathsf{P_0^T} \; \frac{\Phi \vdash \Gamma^*, \mathbf{0}, \Delta}{\Phi \vdash \Gamma^*, \Delta} \qquad \mathsf{P_\invamp^T} \; \frac{\Phi \vdash \Gamma^*, P, Q, \Delta}{\Phi \vdash \Gamma^*, P \invamp Q, \Delta} \qquad \mathsf{P_{sym}^+} \; \frac{\Phi \vdash \Gamma^*, P, Q, \Delta}{\Phi \vdash \Gamma^*, Q, P, \Delta} \qquad \mathsf{P_{wk}^+} \; \frac{\Phi \vdash \Gamma^*, \Delta}{\Phi \vdash \Gamma^*, P, \Delta}$$

$$\mathsf{P_{cut}} \; \frac{\Phi \vdash \Gamma^*, N^\perp, \Gamma_1 \qquad \Phi \vdash N, \Delta^+}{\Phi \vdash \Gamma^*, \Delta^+, \Gamma_1} \qquad\qquad \mathsf{P_{cut}^0} \; \frac{\Phi \vdash N^\perp \qquad \Phi \vdash N, Q}{\Phi \vdash Q}$$

$$\mathsf{P_{id\oslash}} \; \frac{\Phi \vdash N, Q, \Delta^+}{\Phi \vdash M, N, M^\perp \lhd Q, \Delta^+} \qquad\qquad \mathsf{P_{\oplus i}^T} \; \frac{\Phi \vdash \Gamma, P_i, \Delta}{\Phi \vdash \Gamma, P_1 \oplus P_2, \Delta}$$

$$\mathsf{P_{mul}} \; \frac{\Phi \vdash M, \Gamma, \Delta^+ \qquad \Phi \vdash N, \Delta_1^+}{\Phi \vdash M, \Gamma, N, \Delta^+, \Delta_1^+} \qquad\qquad \mathsf{P_{id}} \; \frac{}{\Phi \vdash N, N^\perp} \qquad \mathsf{P_{ana}} \; \frac{\Phi \vdash M, N, N^\perp}{\Phi \vdash !M, N^\perp}$$

$$\mathsf{P_{\multimap}} \; \frac{\Phi \vdash M, \Gamma, P \qquad \Phi \vdash N, \Delta^+}{\Phi \vdash M, \Gamma, P \oslash N, \Delta^+} \qquad\qquad \mathsf{P_{\& i}^T} \; \frac{\Phi \vdash \Gamma, M_1 \& M_2, \Delta}{\Phi \vdash \Gamma, M_i, \Delta}$$

$$\mathsf{P_{der}^!} \; \frac{\Phi \vdash \Gamma, !M, \Delta}{\Phi \vdash \Gamma, M, \Delta} \qquad \mathsf{P_{con}^!} \; \frac{\Phi \vdash \Gamma, !M, \Delta}{\Phi \vdash \Gamma, !M, !M, \Delta} \qquad \mathsf{P_{der}^?} \; \frac{\Phi \vdash \Gamma, P, \Delta}{\Phi \vdash \Gamma, ?P, \Delta} \qquad \mathsf{P_{con}^?} \; \frac{\Phi \vdash \Gamma, ?P, ?P, \Delta}{\Phi \vdash \Gamma, ?P, \Delta}$$

$$\mathsf{P_\forall^T} \; \frac{X; \Theta \vdash \Gamma, \forall x.N, \Delta}{X; \Theta \vdash \Gamma, N[s/x], \Delta} \; FV(s) \subseteq X \qquad\qquad \mathsf{P_\exists^T} \; \frac{X; \Theta \vdash \Gamma, P[s/x], \Delta}{X; \Theta \vdash \Gamma, \exists x.P, \Delta} \; FV(s) \subseteq X$$

The non-core rules of WS1 are given in Figure 5, with Δ^+ ranging over lists of positive formulas, Γ^* over non-empty lists of formulas. These rules reflect some of the categorical structure enjoyed by our games model, and allow straightforward interpretation of other logics and programming languages inside WS1. They include a cut rule, a multiplicative $\otimes$ rule, a restricted form of the exchange rule, weakening, and so on. We will later see that these rules are admissible with respect to the rules in Figures 2, 3 and 4, when restricted to the exponential-free subsystem of WS1. Informally, we can interpret each of these rules as follows:

13

- In the cases of $P_\otimes^\top$, $P_\mathbf{1}^\top$, $P_\otimes^\top$, $P_\mathbf{0}^\top$, P_{sym}^+ and P_{sym}^-, the premise and conclusion are the same game, up to retagging, and can be interpreted using game isomorphisms.

- In the cases of $P_{\&1}^\top$, $P_{\&2}^\top$, P_{wk}^-, $P_{der}^!$ a strategy on the conclusion can be obtained by using only part of the strategy on the premise. For example, for P_{wk}^- we remove all moves in M.

- In the cases of $P_{\oplus1}^\top$, $P_{\oplus2}^\top$, P_{wk}^+, $P_{der}^?$, a strategy on the conclusion can be obtained by using the strategy on the premise and ignoring the extra moves available to Player.

- The P_{id} rule requires a strategy on $N \multimap N$: we can use a *copycat* strategy in which Player always switches component, playing the move that Opponent previously played. The $P_{id\oslash}$ rule can be interpreted by playing copycat in the M component.

- The P_{cut} and P_{cut}^0 rules can be interpreted by playing the two strategies given by the premises against each other in the N component: "parallel composition plus hiding".

- The P_{mul} rule can be interpreted by combining the strategies given by the premises in a multiplicative manner: Opponent's moves in M, Γ are responded to in accordance with the first premise, and moves in N in accordance with the second. The $P_{\multimap}$ rule can be interpreted similarly.

- To interpret $P_{con}^?$, we can construct a strategy on the conclusion by identifying the two copies of $?P$ in the premise. To interpret $P_{con}^!$, we can construct a strategy on the conclusion by identifying the two copies of $!M$ in the conclusion.

- We can interpret P_{ana} using the following construction: given a map $N \multimap M \oslash N$, we may "unwrap" it an infinite number of times to yield a strategy on $N \multimap !M$. The N component represents a parameter that can be used to pass information between the separate threads, to admit history-sensitive behaviour.

3.2.3. Embedding of Intuitionistic Linear Logic

For any negative formulas M, N, define $M \multimap N$ to be $N \lhd M^\perp$. Thus any formula of first-order Intuitionistic Linear Logic is a negative formula of WS1. We sketch an embedding into WS1 of proofs of ILL (over the connectives $\otimes, \multimap, \forall, \&, \mathbf{1}, \perp, !$ and (negative) atoms, formulated with left- and right- introduction rules as in [37]).

Proposition 3.1. *For any proof p of $M_1, \ldots, M_n \vdash N$ in ILL with free variables in X, there is a proof $\kappa(p)$ in WS1 of $X; \emptyset \vdash N, M_1^\perp, \ldots, M_n^\perp$.*

Proof We show that for each rule of ILL there is a derivation in WS1 of the conclusion from the premises.

The left $\otimes$ rule corresponds to $P_\otimes^\top$. For the right $\otimes$ rule, with $\Gamma = G_1, \ldots, G_n$ and $\Delta = D_1, \ldots, D_m$, we duplicate the proof and use P_{mul} as follows:

$$
P_{mul} \cfrac{\vdash N, D_1, \ldots, D_m \qquad \vdash M, G_1, \ldots, G_n}{P_{sym}^+ \cfrac{\vdash N, M, D_1, \ldots, D_m, G_1, \ldots, G_n}{\vdots}}
$$

$$
P_{mul} \cfrac{\vdash M, G_1, \ldots, G_n \qquad \vdash N, D_1, \ldots, D_m}{P_\otimes \cfrac{\vdash M, N, G_1, \ldots, G_n, D_1, \ldots, D_m}{P_{sym}^+ \cfrac{\vdots}{\vdash N, M, G_1, \ldots, G_n, D_1, \ldots, D_m}}}
$$

$$\vdash M \otimes N, G_1, \ldots, G_n, D_1, \ldots, D_m$$

14

The left **1** rule corresponds to $\mathsf{P}^\top_\mathbf{0}$. The right **1** rule corresponds to $\mathsf{P}_\mathbf{1}$. The left $\multimap$ rule can be derived as follows:

$$
\mathsf{P}_{\multimap}\ \mathsf{P}^+_{\mathsf{sym}}\ \cfrac{\cfrac{\vdash L, D_1, \ldots, D_m, N^\perp \qquad \vdash M, G_1, \ldots, G_n}{\vdash L, D_1, \ldots, D_m, N^\perp \oslash M, G_1, \ldots, G_n}}{\mathsf{P}^+_{\mathsf{sym}}\ \cfrac{\vdots}{\vdash L, G_1, \ldots, G_n, N^\perp \oslash M, D_1, \ldots, D_m}}
$$

The right $\multimap$ rule corresponds to $\mathsf{P}_{\lhd}$. The left $\&$ rules correspond to the $\mathsf{P}^\top_\oplus$ rules. The right $\&$ rule corresponds to $\mathsf{P}_\&$. The right-$\forall$ rule corresponds to $\mathsf{P}_\forall$ and the left-$\forall$ rule corresponds to $\mathsf{P}^\top_\exists$.

The dereliction, contraction and weakening rules for the exponential correspond to $\mathsf{P}^?_{\mathsf{der}}$, $\mathsf{P}^?_{\mathsf{con}}$ and $\mathsf{P}^+_{\mathsf{wk}}$ respectively. We next give the translation of the right ! rule (promotion). We first assume Γ consists of a single formula L.

$$
\mathsf{P}_{\mathsf{mul}}\ \mathsf{P}^?_{\mathsf{con}}\ \mathsf{P}_{\mathsf{ana}}\ \cfrac{\cfrac{\cfrac{\vdash N, ?L^\perp \qquad \mathsf{P}_{\mathsf{id}}\ \cfrac{}{\vdash !L, ?L^\perp}}{\vdash N, !L, ?L^\perp, ?L^\perp}}{\vdash N, !L, ?L^\perp}}{\vdash !N, ?L^\perp}
$$

We will later refer to this derived rule as $\mathsf{P}_{\mathsf{prom}}$. If Γ contains more than one formula, we use the equivalence of $!M \otimes !N$ and $!(M \& N)$ in $\mathsf{WS1}$.

The first direction $p_1 \vdash !M \otimes !N \multimap !(M \& N)$ is defined as follows:

$$
\mathsf{P}_\&\ \cfrac{
\mathsf{P}^\top_\otimes\ \mathsf{P}^\top_{⅋}\ \mathsf{P}^!_{\mathsf{der}}\ \mathsf{P}^!_{\mathsf{con}}\ \mathsf{P}_{\mathsf{mul}}\ \cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\mathsf{P}_{\mathsf{id}}\ \cfrac{}{\vdash !M, ?M^\perp} \quad \mathsf{P}_{\mathsf{id}}\ \cfrac{}{\vdash !N, ?N^\perp}}{\vdash !M, !N, ?M^\perp, ?N^\perp}}{\vdash !M, !M, !N, ?M^\perp, ?N^\perp}}{\vdash M, !M, !N, ?M^\perp, ?N^\perp}}{\vdash M, !M, !N, ?M^\perp ⅋ ?N^\perp}}{\vdash M, !M \otimes !N, ?M^\perp ⅋ ?N^\perp}
\qquad
\mathsf{P}^\top_\otimes\ \mathsf{P}^\top_{⅋}\ \mathsf{P}^!_{\mathsf{der}}\ \mathsf{P}_{\mathsf{sym}}\ \mathsf{P}^!_{\mathsf{con}}\ \mathsf{P}_{\mathsf{mul}}\ \cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\mathsf{P}_{\mathsf{id}}\ \cfrac{}{\vdash !M, ?M^\perp} \quad \mathsf{P}_{\mathsf{id}}\ \cfrac{}{\vdash !N, ?N^\perp}}{\vdash !M, !N, ?M^\perp, ?N^\perp}}{\vdash !M, !N, !N, ?M^\perp, ?N^\perp}}{\vdash !N, !M, !N, ?M^\perp, ?N^\perp}}{\vdash N, !M, !N, ?M^\perp, ?N^\perp}}{\vdash N, !M, !N, ?M^\perp ⅋ ?N^\perp}}{\vdash N, !M \otimes !N, ?M^\perp ⅋ ?N^\perp}
}{
\mathsf{P}_{\mathsf{ana}}\ \cfrac{\vdash M \& N, !M \otimes !N, ?M^\perp ⅋ ?N^\perp}{\vdash !(M \& N), ?M^\perp ⅋ ?N^\perp}
}
$$

The second direction $p_2 \vdash !(M \& N) \multimap !M \otimes !N$ is given as follows:

$$
\mathsf{P}_{\mathsf{mul}_\otimes}\ \mathsf{P}^?_{\mathsf{con}}\ \cfrac{\cfrac{
\mathsf{P}_{\mathsf{prom}}\ \mathsf{P}^?_{\mathsf{der}}\ \mathsf{P}^\top_{\oplus} 1\ \cfrac{\cfrac{\cfrac{\mathsf{P}_{\mathsf{id}}\ \cfrac{}{\vdash M, M^\perp}}{\vdash M, M^\perp \oplus N^\perp}}{\vdash M, ?(M^\perp \oplus N^\perp)}}{\vdash !M, ?(M^\perp \oplus N^\perp)}
\qquad
\mathsf{P}_{\mathsf{prom}}\ \mathsf{P}^?_{\mathsf{der}}\ \mathsf{P}^\top_{\oplus} 2\ \cfrac{\cfrac{\cfrac{\mathsf{P}_{\mathsf{id}}\ \cfrac{}{\vdash N, N^\perp}}{\vdash N, M^\perp \oplus N^\perp}}{\vdash N, ?(M^\perp \oplus N^\perp)}}{\vdash !N, ?(M^\perp \oplus N^\perp)}
}{\vdash !M \otimes !N, ?(M^\perp \oplus N^\perp), ?(M^\perp \oplus N^\perp)}}{\vdash !M \otimes !N, ?(M^\perp \oplus N^\perp)}
$$

We can then generalise $\mathsf{P}_{\mathsf{prom}}$ to

$$\frac{\vdash M, ?P_1, ?P_2, \ldots, ?P_{n-1}, ?P_n}{\vdots}$$

$$\mathsf{P}^\mathsf{T}_{\!\!\otimes} \; \frac{\vdash M, ?(P_1 \oplus P_2 \oplus \ldots \oplus P_{n-1}), ?P_n}{\vdash M, ?(P_1 \oplus P_2 \oplus \ldots \oplus P_{n-1}) \otimes ?P_n} \qquad p_2$$

$$\mathsf{P}_{\mathsf{cut}} \; \frac{}{\vdash M, ?(P_1 \oplus P_2 \oplus \ldots \oplus P_{n-1} \oplus P_n)}$$

$$\mathsf{P}_{\mathsf{prom}} \; \frac{}{\vdash !M, ?(P_1 \oplus P_2 \oplus \ldots \oplus P_{n-1} \oplus P_n)}$$

$$\vdots$$

$$\mathsf{P}_{\mathsf{cut}} \; \frac{\vdash !M, ?(P_1 \oplus P_2), \ldots, ?P_{n-1}, ?P_n}{\vdash !M, ?P_1 \otimes ?P_2, \ldots, ?P_{n-1}, ?P_n} \qquad p_1$$

$$\mathsf{P}^\mathsf{T}_{\!\!\otimes} \; \frac{}{\vdash !M, ?P_1, ?P_2, \ldots, ?P_{n-1}, ?P_n}$$

and interpret the right ! rule of ILL. $\qquad\qquad\qquad\qquad\qquad\qquad\Box$

A detailed proof-theoretic analysis of the properties of this translation is beyond the scope of this paper. However, we note that the translation is semantically natural, in the following sense. We shall see in Section 5 that the categorical models of WS1 have (among other properties) the structure of a standard categorical model of ILL: they are *Lafont categories* [32]. The semantics of the quantifier-free fragment of ILL induced by translation into WS1 followed by interpretation in a categorical model coincides with the expected semantics of ILL in a Lafont category.

3.2.4. New Theorems

We next sketch some examples of formulas that are not provable in ILL but are provable in WS1 — i.e. they denote games on which there are uniform winning history-sensitive strategies which are expressible in WS1.

The formulas

$$((A \otimes B \multimap \bot) \otimes (C \otimes D \multimap \bot) \multimap \bot) \multimap$$
$$((A \multimap \bot) \otimes (C \multimap \bot) \multimap \bot) \otimes ((B \multimap \bot) \otimes (D \multimap \bot) \multimap \bot)$$

are not provable, in general, in intuitionistic linear logic (in particular, when A, B, C, D are instantiated as negative atoms). They are a counterpart in ILL of the *medial* rule $[(A \otimes B) \otimes (C \otimes D)] \multimap [(A \otimes C) \otimes (B \otimes D)]$, using an interpretation of depolarised formulas in a polarised setting following [34].

As observed by Blass [7], however, there are (uniform) history-sensitive winning strategies for medial. Informally, suppose:

- Opponent first choses the left hand component in the output (choice 1)

- Opponent then chooses the right hand component in the input (choice 2)

Player can then play copycat in C. If Opponent then switches to the second output component $((B \multimap \bot) \otimes (D \multimap \bot) \multimap \bot)$, Player must enter copycat in D. But this decision relies on knowledge of Opponent's choice 2, which is not possible in an innocent setting and requires history-sensitive knowledge.

An outline WS1 proof of this formula is given in Figure 6. The use of the $\mathsf{P}_\otimes$ demonstrates where the proof branches; there are four branches corresponding to the two uses

16

of $\mathsf{P}_\otimes$. In each of these four branches different $\mathsf{P}_{\otimes i}$ proof rules are chosen at the points labelled $\mathsf{P}_{\otimes 1}$ here.

Similarly, the following theorems of WS1 are not provable in ILL but are provable in WS1:

- $[A \otimes (C\&D)]\&[B \otimes (C\&D)]\&[(A\&B) \otimes C]\&[(A\&B) \otimes D] \multimap$
  $(A\&B) \otimes (C\&D)$, also discussed in [7]

- $\phi_{\mathsf{ex}} \multimap \phi_{\mathsf{ex}} \otimes \phi_{\mathsf{ex}}$ where $\phi_{\mathsf{ex}} = (\phi\&(\phi \multimap \bot)) \multimap \bot$.

3.3. Embedding Polarized Linear Logic in WS1

Polarized Linear Logic (LLP) [27] is a proof system for a polarisation of linear logic into negative and positive formulas. As we have noted, this is entirely different from the polarisation of WS1 formulas employed here: each makes sense within the proof system within which it is defined. Here, we show how proofs of LLP may be represented inside WS1 by translation, with two objectives:

- To clarify the relationship between the two logical systems, and their notions of polarisation.

- To capture both call-by-name and call-by-value λ-calculi via known (and elegant) translations into LLP, which may be composed with our embedding of LLP into WS1. In the call-by-name case, this corresponds with interpretation via intuitionistic linear logic, whereas for call-by-value it is new.

The formulas of LLP (over the units) are as follows:

$$
\begin{array}{rcccccccccccc}
P & ::= & \mathbf{1} & | & \mathbf{0} & | & P \otimes Q & | & P \oplus Q & | & \downarrow N & | & !N \\
N & ::= & \bot & | & \top & | & M \mathbin{\rotatebox[origin=c]{180}{\&}} N & | & M\&N & | & \uparrow P & | & ?P
\end{array}
$$

There is an operation $(-)^\perp$ exchanging polarity, swapping $\mathbf{1}$ for $\bot$, $\mathbf{0}$ for $\top$, $\otimes$ for $\mathbin{\rotatebox[origin=c]{180}{\&}}$, and so on. The presentation of LLP given in [27] omits the linear lifts $\uparrow$ and $\downarrow$ of MALLP. We will include them in our presentation of LLP and its embedding.

A sequent of LLP is a list of LLP formulas. The proof rules for Polarized Linear Logic are given in Figure 7. Γ^- ranges over lists of negative formulas, and Γ' over lists where at most one formula is positive. We say a negative LLP formula N is *reusable* (and write $\mathsf{reuse}(N)$) if every occurrence of $\uparrow$ occurs under a ?. If we exclude the linear lifts $\uparrow$ and $\downarrow$, all negative formulas are reusable. $\mathsf{reuse}(\Gamma^-)$ holds if all formulas in Γ^- are reusable.

Each provable sequent has at most one positive formula, so we can restrict our attention to sequents of this form. It is possible to give semantics to LLP proofs as *innocent* strategies [27], which do not have access to the entire history of play.

We next describe an embedding of LLP inside WS1. Apart from some renaming of units, connectives in LLP will be interpreted by the same connective in WS1. Broadly speaking, positive formulas of LLP will be mapped to negative formulas of WS1, and negative formulas of LLP to positive formulas of WS1. However, under this scheme there is a mismatch for the additives: we will therefore need to map formulas of LLP to *families* of WS1 formulas. The formulas that have a lift as their outermost connective will be mapped to singleton families.

Let $\mathsf{WS1}^-$ denote the set of negative WS1 formulas, and $\mathsf{WS1}^+$ the set of positive WS1 formulas.

17

Figure 6: Outline Proof of Medial

$$\overline{\alpha}, \overline{\gamma} \vdash \top$$
$$\overline{\alpha}, \overline{\gamma} \vdash \top, \overline{\delta}$$
$$\overline{\alpha}, \overline{\gamma} \vdash \overline{\gamma}, \overline{\delta}$$
$$\overline{\alpha}, \overline{\gamma} \vdash \overline{\gamma} ⅋ \overline{\delta}$$
$$\overline{\alpha}, \overline{\gamma} \vdash \bot, (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\gamma} ⅋ \overline{\delta}$$
$$\overline{\alpha} \vdash \gamma, (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\gamma} ⅋ \overline{\delta}$$
$$\overline{\alpha} \vdash \top, \gamma, (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\gamma} ⅋ \overline{\delta}$$
$$\overline{\alpha} \vdash (\top \oslash \gamma), (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\gamma} ⅋ \overline{\delta}$$
$$\overline{\alpha} \vdash (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\gamma} ⅋ \overline{\delta}$$
$$\overline{\alpha} \vdash \bot, \overline{\gamma} ⅋ \overline{\delta}, (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$
$$\overline{\alpha} \vdash (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}), (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$
$$\overline{\alpha} \vdash \top, \overline{\beta}, (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}), (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$

$P_{⅋_1}$
$$\overline{\alpha} \vdash \overline{\alpha}, \overline{\beta}, (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}), (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$
$$\overline{\alpha} \vdash \overline{\alpha} ⅋ \overline{\beta}, (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}), (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$
$$\overline{\alpha} \vdash \overline{\alpha} ⅋ \overline{\beta} \oslash (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}), (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$
$$\overline{\alpha} \vdash \bot, \top \oslash \gamma, (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\alpha} ⅋ \overline{\beta} \oslash (\bot ◁ \overline{\gamma} ⅋ \overline{\delta})$$
$$\vdash \alpha, \top \oslash \gamma, (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\alpha} ⅋ \overline{\beta} \oslash (\bot ◁ \overline{\gamma} ⅋ \overline{\delta})$$
$$\vdash \top, \alpha, \top \oslash \gamma, (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\alpha} ⅋ \overline{\beta} \oslash (\bot ◁ \overline{\gamma} ⅋ \overline{\delta})$$

$P_{⅋_1}$
$$\vdash \top \oslash \alpha, \top \oslash \gamma, (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\alpha} ⅋ \overline{\beta} \oslash (\bot ◁ \overline{\gamma} ⅋ \overline{\delta})$$
$$\vdash (\top \oslash \alpha) ⅋ (\top \oslash \gamma), (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\alpha} ⅋ \overline{\beta} \oslash (\bot ◁ \overline{\gamma} ⅋ \overline{\delta})$$
$$\vdash (\top \oslash \alpha) ⅋ (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \overline{\alpha} ⅋ \overline{\beta} \oslash (\bot ◁ \overline{\gamma} ⅋ \overline{\delta})$$
$$\vdash \bot, \overline{\alpha} ⅋ \overline{\beta}, (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}), (\top \oslash \alpha) ⅋ (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$

$P_{\otimes}$
$$\vdash \bot ◁ \overline{\alpha} ⅋ \overline{\beta}, (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}), (\top \oslash \alpha) ⅋ (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)) \qquad \vdots$$
$$\vdash (\bot ◁ \overline{\alpha} ⅋ \overline{\beta}) \otimes (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}), (\top \oslash \alpha) ⅋ (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$
$$\vdash \top, ((\bot ◁ \overline{\alpha} ⅋ \overline{\beta}) \otimes (\bot ◁ \overline{\gamma} ⅋ \overline{\delta})), (\top \oslash \alpha) ⅋ (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$
$$\vdash \top \oslash ((\bot ◁ \overline{\alpha} ⅋ \overline{\beta}) \otimes (\bot ◁ \overline{\gamma} ⅋ \overline{\delta})), (\top \oslash \alpha) ⅋ (\top \oslash \gamma) \oslash (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta))$$
$$\vdash \bot, (\top \oslash \alpha) ⅋ (\top \oslash \gamma), (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \top \oslash ((\bot ◁ \overline{\alpha} ⅋ \overline{\beta}) \otimes (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}))$$

$P_{\otimes}$
$$\vdash (\bot ◁ (\top \oslash \alpha) ⅋ (\top \oslash \gamma)), (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \top \oslash ((\bot ◁ \overline{\alpha} ⅋ \overline{\beta}) \otimes (\bot ◁ \overline{\gamma} ⅋ \overline{\delta})) \qquad \vdots$$
$$\vdash (\bot ◁ (\top \oslash \alpha) ⅋ (\top \oslash \gamma)) \otimes (\bot ◁ (\top \oslash \beta) ⅋ (\top \oslash \delta)), \top \oslash ((\bot ◁ \overline{\alpha} ⅋ \overline{\beta}) \otimes (\bot ◁ \overline{\gamma} ⅋ \overline{\delta}))$$

$$ax \; \frac{}{\vdash N, N^\perp} \qquad cut \; \frac{\vdash \Gamma, N \qquad \vdash \Delta, N^\perp}{\vdash \Gamma, \Delta} \qquad ex \; \frac{\vdash \Gamma, A, B, \Delta}{\vdash \Gamma, B, A, \Delta}$$

$$\otimes \; \frac{\vdash \Gamma, P \qquad \vdash \Delta, Q}{\vdash \Gamma, \Delta, P \otimes Q} \qquad \invamp \; \frac{\vdash \Gamma, M, N}{\vdash \Gamma, M \invamp N}$$

$$\oplus_1 \; \frac{\vdash \Gamma, P}{\vdash \Gamma, P \oplus Q} \qquad \oplus_2 \; \frac{\vdash \Gamma, Q}{\vdash \Gamma, P \oplus Q} \qquad \& \; \frac{\vdash \Gamma, M \qquad \vdash \Gamma, N}{\vdash \Gamma, M \& N}$$

$$\mathbf{1} \; \frac{}{\vdash \mathbf{1}} \qquad \perp \; \frac{\vdash \Gamma}{\vdash \Gamma, \perp} \qquad \top \; \frac{}{\vdash \Gamma', \top}$$

$$\downarrow \; \frac{\vdash \Gamma^-, N}{\vdash \Gamma^-, \downarrow N} \qquad \uparrow \; \frac{\vdash \Gamma, P}{\vdash \Gamma, \uparrow P}$$

$$! \; \frac{\vdash \Gamma^-, N}{\vdash \Gamma^-, !N} \; \mathsf{reuse}(\Gamma^-) \qquad ?d \; \frac{\vdash \Gamma, P}{\vdash \Gamma, ?P}$$

$$?c \; \frac{\vdash \Gamma, N, N}{\vdash \Gamma, N} \; \mathsf{reuse}(N) \qquad ?w \; \frac{\vdash \Gamma}{\vdash \Gamma, N} \; \mathsf{reuse}(N)$$

Definition A *finite family of negative (resp. positive) WS1 formulas* is a pair (I, f) where I is a finite set and $f : I \to \mathsf{WS1}^-$ (resp. $I \to \mathsf{WS1}^+$).

For brevity, given such a family $F = (I, f)$ we will write $|F|$ for I and F_x for $f(x)$. We will interpret a negative formula of LLP as a finite family of positive WS1 formulas, and a positive formula of LLP as a finite family of negative WS1 formulas. We describe this mapping in Figure 8. Like [34], we decompose the polarity-reversing exponentials of LLP into polarity-preserving exponentials of and polarity-switching linear lifts.

Note that $|i(A^\perp)| = |i(A)|$ and $i(A^\perp)_y = i(A)_y^\perp$. We translate proofs of LLP to families of proofs of WS1 in the following manner:

- Given an LLP proof p of $\vdash N_1, \ldots, N_n$ and $x_i \in |i(N_i)|$ for each i, we construct a proof $i(p, \overrightarrow{x_i})$ of $\vdash \perp, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}$

- Given an LLP proof p of $\vdash N_1, \ldots, N_i, Q, N_{i+1}, \ldots, N_n$ and $x_i \in |i(N_i)|$ for each i, we construct a pair $i(p, \overrightarrow{x_i}) = (y, q)$ where $y \in |i(Q)|$ and q is a proof of $\vdash i(Q)_y, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}$.

Proposition 3.2. *Suppose N is reusable. Then for any x in $|i(N)|$, there is a formula Q and proofs $p \vdash !Q^\perp, i(N)_x$ and $p' \vdash i(N)_x^\perp, ?Q$ such that $[\![p]\!]$ and $[\![p']\!]$ are inverses.*

Proof Simple induction, making use of isomorphisms $!(M \& N) \cong !M \otimes !N$. $\qquad \square$

Proposition 3.3. *For each LLP formula P, $y \in |i(P)|$ and sequence of negative WS1 formulas Δ^- there is a WS1 proof $\mathsf{P}_{P,y}^\top \vdash i(P)_y, \Delta^-, \top$.*

Figure 8: LLP formulas as families of WS1 formulas

$A \in \mathsf{LLP}$	$i(A) \in \mathsf{Fam\ WS1}$
1	$(\{*\}, * \mapsto \mathbf{1})$
0	$(\{*\}, * \mapsto \bot)$
$P \otimes Q$	$(|i(P)| \times |i(Q)|, \langle x, y \rangle \mapsto i(P)_x \otimes i(Q)_y)$
$P \oplus Q$	$(|i(P)| \uplus |i(Q)|, [\mathsf{in}_1(x) \mapsto i(P)_x, \mathsf{in}_2(y) \mapsto i(Q)_y])$
$!N$	$(\{*\}, * \mapsto !\&_{j \in |i(N)|}(\bot \lhd i(N)_j))$
$\downarrow N$	$(\{*\}, * \mapsto \&_{j \in |i(N)|}(\bot \lhd i(N)_j))$
$\bot$	$(\{*\}, * \mapsto \mathbf{0})$
$\top$	$(\{*\}, * \mapsto \top)$
$M \parr N$	$(|i(M)| \times |i(N)|, \langle x, y \rangle \mapsto i(M)_x \parr i(N)_y)$
$M \& N$	$(|i(M)| \uplus |i(N)|, [\mathsf{in}_1(x) \mapsto i(M)_x, \mathsf{in}_2(y) \mapsto i(N)_y])$
$\uparrow P$	$(\{*\}, * \mapsto \bigoplus_{j \in |i(P)|}(\top \oslash i(P)_j))$
$?P$	$(\{*\}, * \mapsto ?\bigoplus_{j \in |i(P)|}(\top \oslash i(P)_j))$

Proof Simple induction on P. □

We next show how each of the LLP proof rules is translated. The translation is simple; we demonstrate some representative cases.

- The *cut* rule, with $p = cut(q, r)$: Suppose $\Gamma = N_1, \ldots, N_i, P, N_{i+1}, \ldots N_n$ and $\Delta = M_1, \ldots, M_m$. Let $x_i \in |i(N_i)|$ and $y_i \in |i(M_i)|$. Then $i(r, \overrightarrow{y_i}) = (y, t)$ with $y \in |i(N^\perp)|$ and $t \vdash i(N^\perp)_y, i(M_1)_{y_1}, \ldots, i(M_n)_{y_n}$. Then $i(q, \overrightarrow{x_i}, y) = (y', q')$ where $y' \in |i(P)|$ and

$$q' \vdash i(P)_{y'}, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}, i(N)_y.$$

Applying $\mathsf{P_{cut}}$ to this proof and t results in a proof g of

$$\vdash i(P)_{y'}, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}, i(M_1)_{y_1}, \ldots, i(M_m)_{y_m}$$

and we set $i(p, \overrightarrow{x_i}, \overrightarrow{y_i}) = (y', g)$.

The case where $\Gamma = N_1, \ldots, N_n$ and $\Delta = M_1, \ldots, M_m$ is similar.

- The $\uparrow$ rule, with $p = \uparrow (q)$: Let $\Gamma = N_1, \ldots, N_n$ and $x_i \in |i(N_i)|$. Then $i(q, \overrightarrow{x_i}) = (y, q)$ where $q \vdash i(P)_y, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}$. We set $i(p, \overrightarrow{x_i})$ to be the following proof:

$$
\mathsf{P}_{\oplus y}
\cfrac{
\cfrac{
\cfrac{
\cfrac{
q \vdash i(P)_y, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}
}{
\vdash \top, i(P)_y, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}
}
}{
\vdash \top \oslash i(P)_y, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}
}
}{
\vdash \bigoplus_{j \in |i(P)|} \top \oslash i(P)_j, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}
}
}{
\cfrac{
\vdash \bot, i(N_1)_{x_1} \parr \ldots \parr i(N_n)_{x_n} \parr (\bigoplus_{j \in |i(P)|} \top \oslash i(P)_j)
}{
\vdash \bot, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}, \bigoplus_{j \in |i(P)|} \top \oslash i(P)_j
}
}
$$

Note that in the semantics of this rule two moves are played: the opening lift overall (O-move) and the opening lift in the derelicted component (P-move), which corresponds to "focusing" on that component.

- The $?c$ rule, with $p = ?c(q)$:

  If $\Gamma = N_1, \ldots, N_n$ and $x_i \in |i(N_i)|$ and $x \in |i(N)|$ then $i(q, \overrightarrow{x_i}, x, x)$ is a proof of $\vdash \bot, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}, i(N)_x, i(N)_x$. We can apply Proposition 3.2 and use $?$-contraction in $\mathsf{WS1}$ to yield a proof q' of
  $\vdash \bot, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}, i(N)_x$ and we set $i(p, \overrightarrow{x_i}, x) = q'$.

  If $\Gamma = N_1, \ldots, N_i, P, N_{i+1}, \ldots, N_n$ and $x_i \in |i(N_i)|$ and $x \in |i(N)|$ then $i(q, \overrightarrow{x_i}, x, x) = (y, q')$ where $q' \vdash i(P)_y, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}, i(N)_x, i(N)_x$. We can apply Proposition 3.2 and use $?$-contraction in $\mathsf{WS1}$ to yield a proof q'' of

  $$\vdash i(P)_y, i(N_1)_{x_1}, \ldots, i(N_n)_{x_n}, i(N)_x$$

  and we set $i(p, \overrightarrow{x_i}, x) = (y, q'')$.

We can hence interpret proofs in LLP as (families of) proofs in $\mathsf{WS1}$.

4. Representing Imperative Programs and their Properties

4.1. Imperative Cell

As an example of a proof of $\mathsf{WS1}$ capturing imperative behaviour (and which does not correspond to a proof of intuitionistic or polarized linear logic), we give a proof which denotes the Boolean reference cell strategy described in Section 2.2, the *cell* strategy of [5].

Recall that this is a strategy for the game $!(\mathbf{B}\&\mathbf{Bi})$, where $\mathbf{B} = \bot \lhd \top \oplus \top$ and $\mathbf{Bi} = (\bot\&\bot) \lhd \top$. We can parametrise the cell by a starting value, yielding a strategy on $\mathbf{B} \multimap !(\mathbf{B}\&\mathbf{Bi})$. We may obtain this strategy using a finite strategy $p : \mathbf{B} \multimap (\mathbf{B}\&\mathbf{Bi}) \oslash \mathbf{B}$. The strategy p is defined as follows, using the naming conventions from Section 2.2:

$$
\begin{array}{ccccccc}
\mathbf{B} & \multimap & (\mathbf{B} & \& & \mathbf{Bi}) & \oslash & \mathbf{B} \\
& & \mathtt{q} & & & & \\
\mathtt{q} & & & & & & \\
b & & & & & & \\
& & b & & & & \\
& & & & & & \mathtt{q} \\
& & & & & & b \\
\hline
& & & \mathtt{in}(b) & & & \\
& & & \mathtt{ok} & & & \\
& & & & & & \mathtt{q} \\
& & & & & & b \\
\end{array}
$$

To obtain the cell strategy, we consider an infinite unwrapping $(\!(\, p \,)\!) : \mathbf{B} \multimap !(\mathbf{B}\&\mathbf{Bi})$, as performed by the semantics of the $\mathsf{P_{ana}}$ rule.

$$\mathbf{B} \multimap (\mathbf{B} \ \& \ \mathbf{Bi}) \quad \oslash \quad \mathbf{B} \multimap (\mathbf{B} \ \& \ \mathbf{Bi}) \quad \oslash \quad ((\mathbf{B} \ \& \ \mathbf{Bi}) \quad \oslash \quad \ldots)$$

$$\mathtt{in}(b)$$
$$\mathtt{in}(b)$$
$$\mathtt{ok}$$
$$\mathtt{ok}$$
$$q$$
$$q$$
$$b$$
$$b$$
$$\vdots$$

We can represent this strategy in our system using the anamorphism rule $\mathsf{P_{ana}}$: we may prove $!(\mathbf{B}\&\mathbf{Bi}), \mathbf{B}^\perp$ by applying this rule to a proof of $(\mathbf{B}\&\mathbf{Bi}), \mathbf{B}, \mathbf{B}^\perp$. To obtain this, we apply the product rule to a pair of proofs:

- $p_{\mathtt{read}}$, of $\mathbf{B}, \mathbf{B}, \mathbf{B}^\perp$, corresponding to a function which reads its argument, returns it *and* propagates it to the next call, and

- $p_{\mathtt{write}}$, of $\mathbf{Bi}, \mathbf{B}, \mathbf{B}^\perp$, corresponding to a function which ignores its argument, accepts a Boolean input value and propagates it to the next call.

In this proof, if a rule is not labelled it is the unique applicable core rule, and some steps are omitted for brevity.

$$
\cfrac{p_{\mathtt{write}} :\vdash (\bot\&\bot)\lhd\top, \bot\lhd(\top\oplus\top), \top\oslash(\bot\&\bot) \qquad p_{\mathtt{read}} :\vdash \bot\lhd(\top\oplus\top), \bot\lhd(\top\oplus\top), \top\oslash(\bot\&\bot)}{\mathsf{P_{ana}}\ \cfrac{\vdash ((\bot\&\bot)\lhd\top)\&(\bot\lhd(\top\oplus\top)), \bot\lhd(\top\oplus\top), \top\oslash(\bot\&\bot)}{\vdash\,!(((\bot\&\bot)\lhd\top)\&(\bot\lhd(\top\oplus\top))), \top\oslash(\bot\&\bot)}}
$$

where $p_{\mathtt{write}}$ is

$$
\cfrac{\mathsf{P_{\oplus_1}}\ \cfrac{\mathsf{P_{\otimes_1}}\ \cfrac{\mathsf{P_{\oplus_1}}\ \cfrac{\cfrac{\overline{\vdash\top}}{\vdash\top,(\top\oslash(\bot\&\bot))}}{\vdash\top\oplus\top,(\top\oslash(\bot\&\bot))}}{\vdash(\top\oplus\top)\otimes(\top\oslash(\bot\&\bot))}}{\ddots}}{\ }\quad
\cfrac{\mathsf{P_{\oplus_2}}\ \cdots}{\ }
$$

and $p_{\mathtt{read}}$ is

$$\mathsf{P}_{\oplus_2} \cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\mathsf{P}_{\oplus_1}\cfrac{\overline{\;\vdash \top\;}}{\vdash \top \oplus \top}}{\vdash \bot, (\top \oplus \top)}\qquad \mathsf{P}_{\oplus_1}\cfrac{\vdash \top, \bot \lhd (\top \oplus \top)}{\vdash \top \oplus \top, \bot \lhd (\top \oplus \top)}}{\vdash \bot, (\top \oplus \top) \oslash (\bot \lhd (\top \oplus \top))}\qquad \mathsf{P}_{\oplus_2}\cfrac{\cfrac{\cfrac{\mathsf{P}_{\oplus_2}\cfrac{\overline{\;\vdash \top\;}}{\vdash \top \oplus \top}}{\vdash \bot, (\top \oplus \top)}}{\vdash \top, \bot \lhd (\top \oplus \top)}}{\vdash \top \oplus \top, \bot \lhd (\top \oplus \top)}\;\;\;\vdash \bot, (\top \oplus \top) \oslash (\bot \lhd (\top \oplus \top))}{\vdash \bot \& \bot, (\top \oplus \top) \oslash (\bot \lhd (\top \oplus \top))}}{\vdash \top, (\bot \& \bot) \lhd ((\top \oplus \top) \oslash (\bot \lhd (\top \oplus \top)))}}{\vdash \top \oslash (\bot \& \bot), (\top \oplus \top) \oslash (\bot \lhd (\top \oplus \top))}}{\vdash ((\top \oplus \top) \oslash (\bot \lhd (\top \oplus \top))) \,\mathbin{⅋}\, (\top \oslash (\bot \& \bot))}}{\vdash \bot, ((\top \oplus \top) \oslash (\bot \lhd (\top \oplus \top))) \,\mathbin{⅋}\, (\top \oslash (\bot \& \bot))}}{\vdash \bot \lhd (\top \oplus \top), \bot \lhd (\top \oplus \top), \top \oslash (\bot \& \bot)}$$

(with $\mathsf{P}_{⅋_2}$ labelling the lower inferences)

We will later give categorical semantics to WS1, and so the above proof provides a categorical account of this Boolean reference cell, using a final coalgebraic property of the exponential.

We may use this proof to interpret declaration of a Boolean reference in either call-by-name or call-by-value settings, by composition (cut) with (the translation of) a term-in-context of the form $\Gamma, x : \mathbf{var} \vdash M : T$. Thus we may translate the recursion-free fragments of *Idealized Algol* [36] and *Reduced ML* over finite datatypes into WS1, for example.

4.2. State Encapsulation

WS1 is more expressive than total, finitary Idealized Algol: for instance, we may use the anamorphism rule to capture structures such as stacks, capable of storing an arbitrarily large amount of data. A generalised programming construct which corresponds to this capability is the *encapsulation* operation which appears as the thread operator in [38], and as the encaps strategy in [30] where it is used for constructing imperative objects in a model based on the same underlying notion of game as used here. The operator has type

$$(s \to (o \times s)) \to s \to (1 \to o).$$

Here s is the type of the object's internal state. The first argument represents an object which takes an explicit state of type s, and returns a value of type o, together with an updated state. The second argument represents an initial state. Encapsulation returns an object of type $1 \to o$ (a "thunk" of type o) in which the state s is encapsulated — i.e. hidden from the environment, but shared between separate invocations of the object. On first invocation (unthunking) the initial state is used as the input state, and thereafter, each fresh call receives the output state from the previous invocation as its input.

We can represent this operation in WS1 using the P_{ana} rule. To do this, we consider a call-by-value interpretation of types. We may translate call-by-value types as positive formulas of LLP: $\phi^+(1) = \mathbf{1}$, $\phi^+(A \times B) = \phi^+(A) \otimes \phi^+(B)$ and $\phi^+(A \to B) = !(\phi^+(A)^\perp \,\mathbin{⅋}\, \uparrow \phi^+(B))^2$. Thus by composition with the embedding of LLP in

---

[2] This is slightly different to the original embedding presented in [27], which uses ? rather than $\uparrow$ in the translation of $\to$, allowing first-class continuations to be interpreted (the $\lambda\mu$-calculus). The translation adopted here is a form of *linear CPS interpretation*.

WS1, we may translate the types o and s as the families of WS-formulas $i \circ \phi^+(s)$ and $i \circ \phi^+(o)$. Let us assume for simplicity, that these are singleton families $\{S\}$ and $\{O\}$ respectively (i.e. s represent products of function types). Then encaps may be translated as a proof of $\vdash \bot, \top \oslash i \circ \phi^+(1 \to o), S^\bot, i \circ \phi^+(s \to (o \times s))^\bot$ — i.e. $\vdash \bot, \top \oslash\, !\uparrow (\mathbf{0}\,\rotatebox[origin=c]{180}{$\&$}\downarrow O), S^\bot, ?\downarrow (S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))$ — as follows:

$$
\begin{array}{l}
\mathsf{P_{cut}} \cfrac{a \qquad \mathsf{P_{ana}} \cfrac{\mathsf{P^?_{con}} \cfrac{\mathsf{P_{mul}} \cfrac{\mathsf{P_{id}} \cfrac{}{\vdash\,!\uparrow(S^\bot\rotatebox[origin=c]{180}{$\&$}\downarrow(O\otimes S)),?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))} \quad b}{\vdash\uparrow(\downarrow O),!\uparrow(S^\bot\rotatebox[origin=c]{180}{$\&$}\downarrow(O\otimes S)),S,?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot)),?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot)),S^\bot}}{\vdash\uparrow(\downarrow O),!\uparrow(S^\bot\rotatebox[origin=c]{180}{$\&$}\downarrow(O\otimes S)),S,?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot)),S^\bot}}{\cfrac{\vdash\uparrow(\downarrow O),!\uparrow(S^\bot\rotatebox[origin=c]{180}{$\&$}\downarrow(O\otimes S))\otimes S,?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))\rotatebox[origin=c]{180}{$\&$}S^\bot}{\vdash\,!\uparrow(\downarrow O),?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))\rotatebox[origin=c]{180}{$\&$}S^\bot}}}{\vdash\,!\uparrow(\mathbf{0}\rotatebox[origin=c]{180}{$\&$}\downarrow O),?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))\rotatebox[origin=c]{180}{$\&$}S^\bot}}
\\[3ex]
\mathsf{P^\top_{\rotatebox[origin=c]{180}{$\&$}}} \cfrac{\vdash\,!\uparrow(\mathbf{0}\rotatebox[origin=c]{180}{$\&$}\downarrow O),?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))\rotatebox[origin=c]{180}{$\&$}S^\bot}{\vdash\,!\uparrow(\mathbf{0}\rotatebox[origin=c]{180}{$\&$}\downarrow O),?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot)),S^\bot}
\\[2ex]
\mathsf{P^+_{sym}} \cfrac{}{\vdash\,!\uparrow(\mathbf{0}\rotatebox[origin=c]{180}{$\&$}\downarrow O),S^\bot,?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))}
\\[2ex]
\cfrac{}{\vdash\top,!\uparrow(\mathbf{0}\rotatebox[origin=c]{180}{$\&$}\downarrow O),S^\bot,?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))}
\\[2ex]
\cfrac{}{\vdash\bot,\top\oslash\,!\uparrow(\mathbf{0}\rotatebox[origin=c]{180}{$\&$}\downarrow O),S^\bot,?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))}
\end{array}
$$

where a is the evident isomorphism $\vdash\,!\uparrow(\mathbf{0}\rotatebox[origin=c]{180}{$\&$}\downarrow O),?\downarrow\uparrow O^\bot$ and b is:

$$
\begin{array}{l}
\mathsf{P^?_{der}} \cfrac{\mathsf{P_{mul\otimes}} \cfrac{\mathsf{P_{id}} \cfrac{}{\vdash S,S^\bot} \quad \mathsf{P^+_{sym}}\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\mathsf{P_{mul}}\cfrac{\mathsf{P_{id}}\cfrac{}{\vdash O,O^\bot}\quad\mathsf{P_{id}}\cfrac{}{\vdash S,S^\bot}}{\vdash O,S,O^\bot,S^\bot}}{\mathsf{P^\top_{\rotatebox[origin=c]{180}{$\&$}}}\cfrac{\vdash O\oslash S,O^\bot,S^\bot}{\vdash O\oslash S,O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot}}}{\vdash\top,O\oslash S,O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot}}{\vdash O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot\rotatebox[origin=c]{180}{$\&$}(\downarrow O\oslash S)}}{\vdash\bot,O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot,\downarrow O\oslash S}}{\vdash S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot),\downarrow O\oslash S,S^\bot}}{\vdash S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot),S^\bot,\downarrow O\oslash S}}{\cfrac{\cfrac{\cfrac{\vdash\top,S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot),\downarrow O\oslash S,S^\bot}{\vdash\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot)),\downarrow O\oslash S,S^\bot}}{\vdash\bot,(\downarrow O\oslash S)\rotatebox[origin=c]{180}{$\&$}\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot))\rotatebox[origin=c]{180}{$\&$}S^\bot}}{\vdash\uparrow(\downarrow O),S,\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot)),S^\bot}}}}{\vdash\uparrow(\downarrow O),S,?\downarrow(S\otimes\uparrow(O^\bot\rotatebox[origin=c]{180}{$\&$}S^\bot)),S^\bot}
\end{array}
$$

4.3. Coroutines

We may also give a proof denoting a *coroutining* operation, permitting a form of deterministic multithreading, defined as a strategy in [23, 24]. In a call-by-name setting, this corresponds to an operation taking two terms s, t of type $\mathbf{com} \to \mathbf{com}$, and returning a command which runs s: when (and if) s calls its argument, control passes to t. When t calls its argument, control is passed back to s, and so on, until either s or t terminates.

We can define a coroutining operator cocomp $\vdash \Sigma, ?(\Sigma^\bot\oslash!\Sigma), ?(\Sigma^\bot\oslash!\Sigma)$, where $\Sigma = \top \lhd \bot$. We first give a proof o of $(!\Sigma \multimap \bot) \multimap\,!\Sigma$.

$$P_{id} \frac{}{\vdash \bot \lhd ?(\top \oslash \bot), \top \oslash !(\bot \lhd \top)}$$
$$\frac{}{\vdash \top, \bot \lhd ?(\top \oslash \bot), \top \oslash !(\bot \lhd \top)}$$
$$\frac{}{\vdash \top \oslash (\bot \lhd ?(\top \oslash \bot)), \top \oslash !(\bot \lhd \top)}$$
$$\frac{}{\vdash \bot, \top \oslash !(\bot \lhd \top), \top \oslash (\bot \lhd ?(\top \oslash \bot))}$$
$$\frac{}{\vdash \bot \lhd \top, !(\bot \lhd \top), \top \oslash (\bot \lhd ?(\top \oslash \bot))}$$
$$\frac{}{\vdash !(\bot \lhd \top), \top \oslash (\bot \lhd ?(\top \oslash \bot))}$$
$$\frac{}{\vdash \top, !(\bot \lhd \top), \top \oslash (\bot \lhd ?(\top \oslash \bot))}$$
$$\frac{}{\vdash \top \oslash !(\bot \lhd \top), \top \oslash (\bot \lhd ?(\top \oslash \bot))}$$
$$\frac{}{\vdash \bot, \top \oslash (\bot \lhd ?(\top \oslash \bot)), \top \oslash !(\bot \lhd \top)}$$
$$\frac{}{\vdash \bot, \top, \bot \lhd ?(\top \oslash \bot), \top \oslash !(\bot \lhd \top)}$$
$$\frac{}{\vdash \bot \lhd \top, \bot \lhd ?(\top \oslash \bot), \top \oslash !(\bot \lhd \top)}$$
$$P_{ana} \frac{}{\vdash !(\bot \lhd \top), \top \oslash !(\bot \lhd \top)}$$
$$\frac{}{\vdash !(\bot \lhd \top) \lhd (\top \oslash !(\bot \lhd \top))}$$

We next define a proof $o' \vdash (!\Sigma \multimap \Sigma) \multimap \bot \multimap !\Sigma$, which connects the output move of the first argument to the Player-move in the second argument.

$$P_{mul\otimes} \frac{\dfrac{}{\vdash \bot, \top} \quad \dfrac{}{\vdash !\Sigma, ?\Sigma^{\perp}}}{\vdash \bot \otimes !\Sigma, ?\Sigma^{\perp}, \top}$$
$$\frac{}{\vdash \top, \bot, !\Sigma, ?\Sigma^{\perp}, \top}$$
$$\frac{}{\vdash (\top \oslash \bot) \oslash !\Sigma, ?\Sigma^{\perp}, \top}$$
$$\frac{}{\vdash \bot, ?\Sigma^{\perp}, \top, (\top \oslash \bot) \oslash !\Sigma}$$
$$P_{cut} \frac{o \vdash !\Sigma, \top \oslash !\Sigma \quad \dfrac{}{\vdash \bot \lhd ?\Sigma^{\perp}, \top, (\top \oslash \bot) \oslash !\Sigma}}{\vdash !\Sigma, \top, (\top \oslash \bot) \oslash !\Sigma}$$
$$\frac{}{\vdash !\Sigma \lhd \top, \Sigma^{\perp} \oslash !\Sigma}$$

We can then define cocomp.

$$P_{cut} \frac{\dfrac{\dfrac{\dfrac{}{\vdash \top}}{\vdash \bot, \top} \quad \dfrac{\dfrac{}{\vdash \top}}{\vdash \bot, \top}}{\vdash \bot \otimes \bot, \top}}{\dfrac{\vdash \top, \bot, \bot, \top}{\dfrac{\vdash (\top \oslash \bot) \oslash \bot, \top}{\dfrac{\vdash \bot, \top, (\top \oslash \bot) \oslash \bot}{\vdash \Sigma, \Sigma^{\perp} \oslash \bot}}}} \quad \cdots}{}$$

4.4. Specifying Properties of Programs

The formulas of WS1 are more expressive than the types of languages such as Idealized Algol, and hence they enable the behaviour of history sensitive strategies to be specified both more abstractly and more precisely. For example, formulas can specify the order in which arguments are interrogated, how many times they are interrogated, and relationships between inputs and outputs of ground type (using the first-order structure).

4.4.1. Data-Independent Programming

We can use quantifiers to represent *data-independent* structures such as cells and stacks, where the underlying ground type at a given $\mathcal{L}$-structure L is $|L|$. As a formula/game, this ground type is represented by $\mathbf{V} = \bot \lhd \exists x.\top$ — a dialogue in this game consists of Opponent playing a question move q and Player responding with an element of $|L|$. We can represent a stream of such values using the formula $!\mathbf{V}$.

Let $\mathbf{Vi} = \forall x.\bot \lhd \top$ represent an 'input version' of $\mathbf{V}$, where Opponent plays an $|L|$ value and Player then accepts it, analogous to $\mathbf{Bi}$ above. The type of a stack object can then be given by the formula $!(\mathbf{V}\&\mathbf{Vi})$, with a "pop" and a "push" method. We give a proof denoting the behaviour of such a stack, parametrised by a starting stack, of type $!\mathbf{V} \multimap !(\mathbf{V}\&\mathbf{Vi})$.

$$
\mathsf{P_{ana}} \cfrac{
\mathsf{P^!_{con}}\mathsf{P^!_{der}} \cfrac{
\mathsf{P_{id}} \cfrac{}{\vdash !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}
}{
\cfrac{\vdash !(\bot \lhd \exists x.\top), !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{\vdash \bot \lhd \exists x.\top, !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}
}
\qquad
\mathsf{P^x_{\exists}} \cfrac{
\mathsf{P_{id}} \cfrac{}{\{x\};\vdash !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}
}{
\cfrac{\{x\};\vdash \top, !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{
\cfrac{\{x\};\vdash \exists x.\top, !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{
\cfrac{\{x\};\vdash \bot, \exists x.\top, !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{
\cfrac{\{x\};\vdash \bot \lhd \exists x.\top, !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{
\cfrac{\{x\};\vdash !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{
\cfrac{\{x\};\vdash \top, !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{
\cfrac{\{x\};\vdash \bot, \top, !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{
\cfrac{\vdash \forall x.\bot, \top, !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{\vdash \forall x.\bot \lhd \top, !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}
}}}}}}}
}
}{
\cfrac{\vdash (\bot \lhd \exists x.\top)\&(\forall x.\bot \lhd \top), !(\bot \lhd \exists x.\top), ?(\top \oslash \forall x.\bot)}{\vdash !((\bot \lhd \exists x.\top)\&(\forall x.\bot \lhd \top)), ?(\top \oslash \forall x.\bot)}
}
$$

Once again, we use $\mathsf{P_{ana}}$ to obtain the infinite behaviour, applied to a proof q of $!\mathbf{V} \multimap (\mathbf{V}\&\mathbf{Vi})\oslash!\mathbf{V}$. The strategy denoted by q performs as 'copycat' in the $!\mathbf{V} \multimap \mathbf{V}\oslash!\mathbf{V}$ component, and in the $!\mathbf{V} \multimap \mathbf{Vi}\oslash!\mathbf{V}$ component behaves as follows:

$$
\begin{array}{cccc}
!\mathbf{V} & \multimap & \mathbf{Vi} & \oslash & !\mathbf{V} \\
 & & \texttt{in}(v) & & \\
 & & \texttt{ok} & & \\
 & & & & \texttt{q} \\
 & & & & v \\
\end{array}
$$

and then enters copycat.

4.4.2. Good Variables

One respect in which the game semantics of Idealized Algol (and other imperative languages) fails to reflect its syntax fully is in the existence in the model of *bad variables* which do not return the last value assigned to them [5]. In $\mathsf{WS1}$ we may define formulas for which the only proof denotes a good variable.

The formula $\mathbf{worm} = \mathbf{Bi} \oslash !\mathbf{B}$ represents a Boolean variable which can be written once, then read many times. One proof/strategy of this formula will indeed be a valid Boolean cell: if Opponent plays $\texttt{inputX}$ then Player responds with $\texttt{ok}$, if Opponent then tries to read the cell $\texttt{q}$, then Player responds with $\texttt{X}$. But there are also bad variables: for example, the read method may always return $\texttt{True}$ regardless of what was written.

To exclude such behaviour, we can replace the input/output moves with atoms. Define $\mathbf{B}^{\phi,\psi} = \bot \lhd (\overline{\phi} \oplus \overline{\psi})$ and $\mathbf{Bi}^{\phi,\psi} = (\phi \& \psi) \lhd \top$, with $\mathbf{worm}^{\phi,\psi} = \mathbf{Bi}^{\phi,\psi} \oslash !\mathbf{B}^{\phi,\psi}$. If ϕ and ψ are assigned $\mathtt{tt}$, then this denotes the same dialogue as $\mathbf{worm}$. However, the denotation of any proof of $\mathbf{worm}^{\phi,\psi}$ at such a model must be the good variable strategy. The rule for ϕ (and semantically, uniformity of strategies) ensures that ϕ must be played before $\overline{\phi}$, and ψ before $\overline{\psi}$. Consequently, Player can only respond with a particular Boolean value in the $\mathtt{read}$ component if that same value has previously been given as an input in the $\mathtt{write}$ component, so good-variable behaviour is assured. The following proof of this formula uses only the core rules and the promotion rule.

$$
\mathsf{P}_{\oplus 1} \quad
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{\overline{\phi} \vdash \top}{\overline{\phi} \vdash \overline{\phi}}
}{\overline{\phi} \vdash \overline{\phi} \oplus \overline{\psi}}
}{\overline{\phi} \vdash \bot, \overline{\phi} \oplus \overline{\psi}}
}{\overline{\phi} \vdash \bot \lhd \overline{\phi} \oplus \overline{\psi}}
}{\overline{\phi} \vdash !(\bot \lhd \overline{\phi} \oplus \overline{\psi})} \; \mathsf{prom}
}{\overline{\phi} \vdash \top, !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{\overline{\phi} \vdash \top \oslash !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{\overline{\phi} \vdash \bot, \top \oslash !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{\overline{\phi} \vdash \bot, \top, !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{\vdash \phi, \top, !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
$$

$$
\mathsf{P}_{\oplus 2} \quad
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{
\frac{\overline{\psi} \vdash \top}{\overline{\psi} \vdash \overline{\psi}}
}{\overline{\psi} \vdash \overline{\phi} \oplus \overline{\psi}}
}{\overline{\psi} \vdash \bot, \overline{\phi} \oplus \overline{\psi}}
}{\overline{\psi} \vdash \bot \lhd \overline{\phi} \oplus \overline{\psi}}
}{\overline{\psi} \vdash !(\bot \lhd \overline{\phi} \oplus \overline{\psi})} \; \mathsf{prom}
}{\overline{\psi} \vdash \top, !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{\overline{\psi} \vdash \top \oslash !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{\overline{\psi} \vdash \bot, \top \oslash !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{\overline{\psi} \vdash \bot, \top, !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{\vdash \psi, \top, !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
$$

$$
\frac{
\frac{
\frac{\vdash (\phi \& \psi), \top, !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}{\vdash (\phi \& \psi) \lhd \top, !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{\vdash ((\phi \& \psi) \lhd \top) \oslash !(\bot \lhd \overline{\phi} \oplus \overline{\psi})}
}{}
$$

We cannot use ! to obtain a formula which admits only an arbitrarily reusable 'good variable', but we can obtain finite approximations. For example, the formula

$$\mathbf{Bi}^{\alpha,\beta} \oslash (\mathbf{worm}^{\phi,\psi} \; \& \; (\mathbf{B}^{\alpha,\beta} \oslash \mathbf{worm}^{\phi,\psi}) \; \& \; (\mathbf{B}^{\alpha,\beta} \oslash (\mathbf{B}^{\alpha,\beta} \oslash \mathbf{worm}^{\phi,\psi})))$$

models a good variable that can be written to twice, and can be read at most twice before the second write. Strategies on such formulas then approximate our reusable cell strategy above on $!(\mathbf{B} \& \mathbf{Bi})$.

5. Categorical Semantics for WS1

To give a formal semantics for our logic, we first introduce a notion of categorical model which captures everything except the first-order structure (quantifiers and atoms). We shall use notation $\eta : F \Rightarrow G : \mathcal{C} \to \mathcal{D}$ to mean η is a natural transformation from F to G with $F, G : \mathcal{C} \to \mathcal{D}$.

First, we define some categories of games that will form the intended instance of our categorical model. Objects in these categories will be negative games, and an arrow $A \to B$ will be a strategy on $A \multimap B$. We can compose strategies using "parallel composition plus hiding". Suppose $\sigma : A \multimap B$ and $\tau : B \multimap C$, define

$$\sigma \| \tau = \{s \in (M_A + M_B + M_C)^* : s|_1 \in P_A \wedge s|_2 \in P_B \wedge s|_3 \in P_C\}$$

and set
$$\tau \circ \sigma = \{s|_{1,3} : s \in \sigma \| \tau\}.$$
It is well-known that $\tau \circ \sigma$ is a well-formed strategy on $A \multimap C$ (see e.g. [3]).

Proposition 5.1. *Composition is associative, and there is an identity $A \to A$ given by the copycat strategy: $\{s \in P_{A \multimap A} : \gamma(s)\}$ where $\gamma(s)$ holds if and only if $t|_1 = t|_2$ for all even-length prefixes t of s.*

Definition The category $\mathcal{G}$ has negative games as objects, and a map $\sigma : A \to B$ is a strategy on $A \multimap B$ with composition and identity as above.

This category has been studied extensively in e.g. [25, 11, 30], and has equivalent presentations using graph games [16] and locally Boolean domains [22].

If A, B and C are bounded, $\sigma : A \multimap B$ and $\tau : B \multimap C$ are total then $\tau \circ \sigma$ is also total. Total strategies do not compose for unbounded games, however. Winning strategies on unbounded games do compose [15], and the identity strategy is winning.

Definition The category $\mathcal{W}$ has negative games as objects and winning strategies as maps.

A map $\sigma : A \to B$ is *strict* if it responds to Opponent's first move with a move in A, if it responds at all. Strict strategies are closed under composition and the identity is strict.

Definition The category $\mathcal{G}_s$ has negative games as objects and strict strategies as maps. The category $\mathcal{W}_s$ has negative games as objects and strict winning strategies as maps.

Isomorphisms in $\mathcal{W}$ correspond to forest isomorphisms and all isomorphisms are total and strict [28].

Each of the above categories can be endowed with symmetric monoidal structure, given by $(I, \otimes)$ where I is the empty game **1** and the action of $\otimes$ on objects is as defined in Section 2.1.

5.1. Sequoidal Closed Structure

The notions of *sequoidal category* and *sequoidal closed category* were first introduced in [21].

Definition A *sequoidal category* consists of:

- A symmetric monoidal category $(\mathcal{C}, I, \otimes)$ (we will call the relevant isomorphisms $\mathsf{assoc} : (A \otimes B) \otimes C \cong A \otimes (B \otimes C)$, $\mathsf{lunit}_\otimes : I \otimes A \cong A$, $\mathsf{runit}_\otimes : A \otimes I \cong I$ and $\mathsf{sym} : A \otimes B \cong B \otimes A$)

- A category $\mathcal{C}_s$

- A right-action $\oslash$ of $\mathcal{C}$ on $\mathcal{C}_s$. That is, a functor $\_ \oslash \_ : \mathcal{C}_s \times \mathcal{C} \to \mathcal{C}_s$ with natural isomorphisms $\mathsf{unit}_\oslash : A \oslash I \cong A$ and $\mathsf{pasc} : A \oslash (B \otimes C) \cong (A \oslash B) \oslash C$ satisfying

the following coherence conditions [17]:

$$A \oslash (B \otimes (C \otimes D)) \xrightarrow{\mathsf{pasc}} (A \oslash B) \otimes (C \otimes D) \xrightarrow{\mathsf{pasc}} ((A \oslash B) \oslash C) \oslash D$$

with $\mathsf{id} \oslash \mathsf{assoc}$ down the left, $\mathsf{pasc} \oslash \mathsf{id}$ on the diagonal, and

$$A \oslash ((B \otimes C) \otimes D) \xrightarrow{\mathsf{pasc}} (A \oslash (B \otimes C)) \oslash D$$

$$A \oslash (I \otimes B) \xrightarrow{\mathsf{pasc}} (A \oslash I) \oslash B$$

with $\mathsf{id} \oslash \mathsf{lunit}_\otimes$ on the left, $\mathsf{unit}_\oslash \oslash \mathsf{id}$ on the diagonal, to $A \oslash B$.

$$A \oslash (B \otimes I) \xrightarrow{\mathsf{pasc}} (A \oslash B) \oslash I$$

with $\mathsf{id} \oslash \mathsf{runit}_\otimes$ on the left, $\mathsf{unit}_\oslash$ on the diagonal, to $A \oslash B$.

- A functor $J : \mathcal{C}_s \to \mathcal{C}$
- A natural transformation $\mathsf{wk} : J(\_) \otimes \_ \Rightarrow J(\_ \oslash \_)$ satisfying further coherence conditions [21]:

$$JA \otimes I \xrightarrow{\mathsf{runit}_\otimes} JA$$

with wk down the left, $J(\mathsf{unit}_\oslash)$ on the diagonal, to $J(A \oslash I)$.

$$(JA \otimes B) \otimes C \xrightarrow{\mathsf{wk} \otimes \mathsf{id}} J(A \oslash B) \otimes C \xrightarrow{\mathsf{wk}} J((A \oslash B) \oslash C)$$

with assoc down the left, $J(\mathsf{pasc})$ on the diagonal, and

$$JA \otimes (B \otimes C) \xrightarrow{\mathsf{wk}} J(A \oslash (B \otimes C))$$

Definition An *inclusive sequoidal category* is a sequoidal category in which $\mathcal{C}_s$ is a full-on-objects subcategory of $\mathcal{C}$ containing wk and the monoidal isomorphisms; J is the inclusion functor; and J reflects isomorphisms.

We can identify this structure in our categories of games: we can extend the left-merge operator $\oslash$ to an action $\mathcal{G}_s \times \mathcal{G} \to \mathcal{G}_s$. If $\sigma : A \to B$ and $\tau : C \to D$ then $\sigma \oslash \tau : A \oslash C \to B \oslash D$ plays as σ between A and B and as τ between C and D. The strictness of σ guarantees that this yields a valid strategy on $(A \oslash C) \multimap (B \oslash D)$. The isomorphisms pasc and $\mathsf{unit}_\oslash$ exist, and there is a natural copycat strategy $\mathsf{wk} : M \otimes N \to M \oslash N$ in $\mathcal{G}_s$, all satisfying the required axioms [24]. The functor J reflects isomorphisms as the inverse of strict isomorphisms are strict. Thus $(\mathcal{G}, \mathcal{G}_s)$ forms an inclusive sequoidal category; as does $(\mathcal{W}, \mathcal{W}_s)$.

Definition An inclusive sequoidal category is *Cartesian* if $\mathcal{C}_s$ has finite products preserved by J (we will write t_A for the unique map $A \to 1$). It is *decomposable* if the natural transformations $\mathsf{dec} = \langle \mathsf{wk}, \mathsf{wk} \circ \mathsf{sym} \rangle : A \otimes B \Rightarrow (A \oslash B) \times (B \oslash A) : \mathcal{C}_s \times \mathcal{C}_s \to \mathcal{C}_s$ and $\mathsf{dec}^0 = t_I : I \Rightarrow 1 : \mathcal{C}_s$ are isomorphisms (so, in particular, $(\mathcal{C}, \otimes, I)$ is an affine SMC).

A Cartesian sequoidal category is *distributive* if the natural transformations $\mathsf{dist} = \langle \pi_1 \oslash \mathsf{id}_C, \pi_2 \oslash \mathsf{id}_C \rangle : (A \times B) \oslash C \Rightarrow (A \oslash C) \times (B \oslash C) : \mathcal{C}_s \times \mathcal{C}_s \times \mathcal{C} \to \mathcal{C}_s$ and $\mathsf{dist}_0 = t_{1 \oslash C} : 1 \oslash C \Rightarrow 1 : \mathcal{C} \to \mathcal{C}_s$ are isomorphisms.

29

We write $\mathsf{dist}^0 : I \oslash C \cong I$ for the isomorphism $(\mathsf{dec}^0)^{-1} \circ \mathsf{dist}_0 \circ (\mathsf{dec}^0 \oslash \mathsf{id})$.

In the game categories defined above, $M \& N$ is a product of M and N, and the empty game I is a terminal object as well as the monoidal unit. The decomposability and distributivity isomorphisms above exist as natural copycat morphisms [24]. In fact, $\mathcal{W}$ and $\mathcal{G}$ have all small products, following the construction in Section 2.1, with the corresponding distributivity isomorphism with respect to $\oslash$.

Definition A *sequoidal closed category* is an inclusive sequoidal category where $\mathcal{C}$ is symmetric monoidal closed and the map $f \mapsto \Lambda(f \circ \mathsf{wk})$ defines a natural isomorphism $\Lambda_s : \mathcal{C}_s(B \oslash A, C) \Rightarrow \mathcal{C}_s(B, A \multimap C)$.

We can show that $\mathcal{G}$ and $\mathcal{W}$ are sequoidal closed, with the internal hom given by $\multimap$ [24].

In any sequoidal closed category, define $\mathsf{app}_s : (A \multimap B) \oslash A \to B$ as $\Lambda_s^{-1}(\mathsf{id})$, and $\mathsf{app} : (A \multimap B) \otimes A \to B = \Lambda^{-1}(\mathsf{id})$, noting that $\mathsf{app} = \mathsf{app}_s \circ \mathsf{wk}$. If $f : A \to B$ let $\Lambda_I(f) : I \to A \multimap B$ denote the name of f, i.e. $\Lambda(f \circ \mathsf{runit}_\otimes)$. Write Λ_I^{-1} for the inverse operation.

Proposition 5.2. *In any sequoidal closed category, $\multimap$ restricts to a functor $\mathcal{C}^{\mathsf{op}} \times \mathcal{C}_s \to \mathcal{C}_s$ with natural isomorphisms $\mathsf{unit}_{\multimap} : I \multimap A \cong A$ and $\mathsf{pasc}_{\multimap} : A \otimes B \multimap C \cong A \multimap (B \multimap C)$ in $\mathcal{C}_s$.*

Proof We need to show that if g is in $\mathcal{C}_s$ then $f \multimap g$ is in $\mathcal{C}_s$. But $f \multimap g = \Lambda(g \circ \mathsf{app} \circ (\mathsf{id} \otimes f)) = \Lambda(g \circ \mathsf{app}_s \circ \mathsf{wk} \circ (\mathsf{id} \otimes f)) = \Lambda(g \circ \mathsf{app}_s \circ (\mathsf{id} \oslash f) \circ \mathsf{wk}) = \Lambda_s(g \circ \mathsf{app}_s \circ (\mathsf{id} \oslash f))$ which is in $\mathcal{C}_s$.

In any symmetric monoidal category the isomorphisms $\mathsf{unit}_{\multimap}$ and $\mathsf{pasc}_{\multimap}$ exist, but we must show that they are strict.

- $\mathsf{unit}_{\multimap} : I \multimap A \to A$ is given by $\mathsf{app} \circ \mathsf{runit}_\otimes^{-1}$. This $\mathsf{app}_s \circ \mathsf{wk} \circ \mathsf{runit}_\otimes^{-1} = \mathsf{app}_s \circ \mathsf{unit}_\oslash^{-1}$ which is a map in $\mathcal{C}_s$.

- $\mathsf{pasc}_{\multimap} : A \otimes B \multimap C \cong A \multimap (B \multimap C)$ is given by $\Lambda(\Lambda(\mathsf{app} \circ \mathsf{assoc})) = \Lambda(\Lambda(\mathsf{app}_s \circ \mathsf{wk} \circ \mathsf{assoc})) = \Lambda(\Lambda(\mathsf{app}_s \circ \mathsf{pasc}^{-1} \circ \mathsf{wk} \circ (\mathsf{wk} \otimes \mathsf{id}))) = \Lambda(\Lambda(\mathsf{app}_s \circ \mathsf{pasc}^{-1} \circ \mathsf{wk}) \circ \mathsf{wk}) = \Lambda_s(\Lambda_s(\mathsf{app}_s \circ \mathsf{pasc}^{-1}))$ which is in $\mathcal{C}_s$.

The inverses of the above maps are strict as J reflects isomorphisms. $\square$

In distributive, decomposable sequoidal closed categories we can also define the following natural transformations:

- The isomorphism $\mathsf{psym} : (A \oslash B) \oslash C \cong (A \oslash C) \oslash B$ given by $\mathsf{pasc} \circ (\mathsf{id} \oslash \mathsf{sym}) \circ \mathsf{pasc}^{-1}$.

- The isomorphism $\mathsf{psym}_{\multimap} : C \multimap (B \multimap A) \cong B \multimap (C \multimap A)$ given by $\mathsf{pasc}_{\multimap} \circ (\mathsf{sym} \multimap \mathsf{id}) \circ \mathsf{pasc}_{\multimap}^{-1}$

- The isomorphism $\mathsf{dist}_{\multimap} : A \multimap (B \times C) \to (A \multimap B) \times (A \multimap C)$ given by $\langle \mathsf{id} \multimap \pi_1, \mathsf{id} \multimap \pi_2 \rangle$, whose inverse is $\Lambda\langle \mathsf{app} \circ (\pi_1 \otimes \mathsf{id}), \mathsf{app} \circ (\pi_2 \otimes \mathsf{id}) \rangle$. This isomorphism exists in any monoidal closed category with products.

- The map $\mathsf{af} : A \Rightarrow I$ given by $(\mathsf{dec}^0)^{-1} \circ \mathsf{t}_A$.

- The isomorphism $\mathsf{dist}^0_{\multimap} : A \multimap I \to I$ given by af whose inverse is $\Lambda(\mathsf{runit}_\otimes \circ (\mathsf{id} \otimes \mathsf{af}))$. We must check that these are inverses: $\mathsf{af} \circ \Lambda(\mathsf{runit}_\otimes \circ (\mathsf{id} \otimes \mathsf{af})) = \mathsf{id}$ as both are maps into the terminal object, and $\Lambda(\mathsf{runit}_\otimes \circ (\mathsf{id} \otimes \mathsf{af})) \circ \mathsf{af} = \Lambda(\mathsf{runit}_\otimes \circ (\mathsf{af} \otimes \mathsf{id}) \circ (\mathsf{af} \otimes \mathsf{id})) = \Lambda(\mathsf{app}) = \mathsf{id}$ as required. We know that $\mathsf{runit}_\otimes \circ (\mathsf{af} \otimes \mathsf{id}) \circ (\mathsf{af} \otimes \mathsf{id}) = \mathsf{app}$ as both are maps into the terminal object.

We can use the structure described above to model the negative connectives of $\mathsf{WS1}$. We will represent positive connectives indirectly, inspired by the fact that strategies on the positive game P correspond to strategies on the negative game $\uparrow P = P^\perp \multimap o$ where o is the one-move game $\perp$. The object o satisfies a special property: an internalised version of *linear functional extensionality* [1].

Definition An object o in a sequoidal closed category satisfies *linear functional extensionality* if the natural transformation $\mathsf{lfe} : (B \multimap o) \oslash A \Rightarrow (A \multimap B) \multimap o : \mathcal{C} \times \mathcal{C}^{\mathsf{op}} \to \mathcal{C}_s$ given by $\Lambda_s(\mathsf{app}_s \circ (\mathsf{id} \oslash \mathsf{app}) \circ (\mathsf{id} \oslash \mathsf{sym}) \circ \mathsf{pasc}^{-1})$ is an isomorphism.

The linear functional extensionality property is characteristic of our *history sensitive, locally alternating* games model [24]: it does not hold in other sequoidal closed categories (e.g. Conway games [21]).

Using linear functional extensionality we can give a natural isomorphism $\mathsf{abs} : o \oslash A \cong o$ by noticing that $o \oslash A \cong (I \multimap o) \oslash A \cong (A \multimap I) \multimap o \cong I \multimap o \cong o$, and thus setting $\mathsf{abs} = \mathsf{unit}_{\multimap} \circ ((\mathsf{dist}^0_{\multimap})^{-1} \multimap \mathsf{id}) \circ \mathsf{lfe} \circ (\mathsf{unit}^{-1}_{\multimap} \oslash \mathsf{id})$.

5.2. Coalgebraic Exponential Comonoid

We next consider the categorical status of the exponential operator $!$. We interpret the core introduction rules for the exponentials, and the key anamorphism rule, by requiring that it is the carrier for a *final coalgebra* of the functor $X \mapsto N \oslash X$.

Recall that a coalgebra for a functor $F : \mathcal{C} \to \mathcal{C}$ is an object A and a map $A \to F(A)$. A *final coalgebra* is a terminal object in the category of coalgebras, that is a coalgebra $\alpha : Z \to F(Z)$ such that for any $f : A \to F(A)$ there is a unique $\llparenthesis f \rrparenthesis : A \to Z$ such that $\alpha \circ \llparenthesis f \rrparenthesis = F(\llparenthesis f \rrparenthesis) \circ f$.

$$
\begin{array}{ccc}
A & \xrightarrow{\ f\ } & F(A) \\
{\scriptstyle \llparenthesis f \rrparenthesis}\big\downarrow & & \big\downarrow{\scriptstyle F(\llparenthesis f \rrparenthesis)} \\
Z & \xrightarrow[\ \alpha\ ]{} & F(Z)
\end{array}
$$

We call $\llparenthesis f \rrparenthesis$ the *anamorphism* of f. Note in particular that if (Z, α) is a final coalgebra for F, then α is an isomorphism, with inverse $\alpha^{-1} = \llparenthesis F(\alpha) \rrparenthesis$.

In $\mathcal{W}$ we define a coalgebra $(!N, \alpha)$ by taking $\alpha : !N \to N \oslash !N$ to be the evident copycat strategy which relabels $\mathsf{in}_1(a)$ on the right to $(a, 1)$ on the left and $\mathsf{in}_2(a, n)$ on the right to $(a, n + 1)$ on the left.

Proposition 5.3. $(!N, \alpha)$ *is the final coalgebra of the functor* $N \oslash \_$ *in the category* $\mathcal{G}$.

Proof Let $\sigma : M \to N \oslash M$. Define $⟬\, \sigma\, ⟭_n : M \to (N \oslash \_)^n(M)$ by $⟬\, \sigma\, ⟭_0 = \mathsf{id}$ and $⟬\, \sigma\, ⟭_{n+1} = (\mathsf{id} \oslash \_)^n(\sigma) \circ ⟬\, \sigma\, ⟭_n$.

$$M \xrightarrow{\;⟬\, \sigma\, ⟭_n\;} (N \oslash \_)^n(M) \xrightarrow{\;(\mathsf{id} \oslash \_)^n(\sigma)\;} (N \oslash \_)^n(N \oslash M) = (N \oslash \_)^{n+1}(M)$$

The strategy $⟬\, \sigma\, ⟭_n$ is a partial approximant to $⟬\, \sigma\, ⟭ : M \to \,! N$. We can show by induction on n that $⟬\, \sigma\, ⟭_{n+1} = (\mathsf{id} \oslash ⟬\, \sigma\, ⟭_n) \circ \sigma$. Similarly, we can define $\alpha_k : \,! N \cong (N \oslash \_)^k(! N) : \alpha_k^{-1}$ by performing the above construction on α. Consider the sequence of maps $M \to \,! N$ defined by $s_k = \alpha_k^{-1} \circ (\mathsf{id} \oslash \_)^k(\epsilon) \circ ⟬\, \sigma\, ⟭_k$ for $k \in \omega$. We can show that $s_{k+1} \sqsupseteq s_k$ by induction on k, and so (s_k) is a chain. Set $⟬\, \sigma\, ⟭ = \bigsqcup \alpha_k^{-1} \circ (\mathsf{id} \oslash \_)^k(\epsilon) \circ ⟬\, \sigma\, ⟭_k$, where ϵ is the empty strategy. It is well-known that $\mathcal{G}$ is cpo-enriched with bottom element ϵ [24].

We wish to show that $⟬\, \sigma\, ⟭$ is the unique strategy such that $\alpha \circ ⟬\, \sigma\, ⟭ = (\mathsf{id} \oslash ⟬\, \sigma\, ⟭) \circ \sigma$. To show that the equation holds, note that $\alpha \circ ⟬\, \sigma\, ⟭ = \alpha \circ \bigsqcup \alpha_k^{-1} \circ (\mathsf{id} \oslash \_)^k(\epsilon) \circ ⟬\, \sigma\, ⟭_k = \alpha \circ \bigsqcup \alpha_{k+1}^{-1} \circ (\mathsf{id} \oslash \_)^{k+1}(\epsilon) \circ ⟬\, \sigma\, ⟭_{k+1} = \bigsqcup \alpha \circ \alpha_{k+1}^{-1} \circ (\mathsf{id} \oslash \_)^{k+1}(\epsilon) \circ ⟬\, \sigma\, ⟭_{k+1} = \bigsqcup (\mathsf{id} \oslash \alpha_k^{-1}) \circ (\mathsf{id} \oslash (\mathsf{id} \oslash \_)^k(\epsilon)) \circ (\mathsf{id} \oslash ⟬\, \sigma\, ⟭_k) \circ \sigma = (\mathsf{id} \oslash \bigsqcup (\alpha_k^{-1} \circ (\mathsf{id} \oslash \_)^k(\epsilon) \circ ⟬\, \sigma\, ⟭_k)) \circ \sigma = (\mathsf{id} \oslash ⟬\, \sigma\, ⟭) \circ \sigma$.

For uniqueness, suppose that $\gamma : M \to \,! N$ is such that $\alpha \circ \gamma = (\mathsf{id} \oslash \gamma) \circ \sigma$. We wish to show that $\gamma = ⟬\, \sigma\, ⟭ = \bigsqcup \alpha_k^{-1} \circ (\mathsf{id} \oslash \_)^k(\epsilon) \circ ⟬\, \sigma\, ⟭_k$. To see that $\gamma \sqsupseteq ⟬\, \sigma\, ⟭$, it suffices to show that γ is an upper bound of the chain, i.e. $\gamma \sqsupseteq \alpha_k^{-1} \circ (\mathsf{id} \oslash \_)^k(\epsilon) \circ ⟬\, \sigma\, ⟭_k$ for each k. This can be shown using a simple induction on k. To see that $\gamma \sqsubseteq ⟬\, \sigma\, ⟭$, we show that each play in γ is also in $⟬\, \sigma\, ⟭$. Consider a play $s \in \gamma : M \to \,! N$. Since s is finite, it must visit only a finite number of copies of N — say, k copies. Then s is also a play in $\alpha_k^{-1} \circ (\mathsf{id} \oslash \_)^k(\epsilon) \circ \alpha_k \circ \gamma$.

It is thus sufficient to show that $(\mathsf{id} \oslash \_)^k(\epsilon) \circ \alpha_k \circ \gamma \sqsubseteq (\mathsf{id} \oslash \_)^k(\epsilon) \circ ⟬\, \sigma\, ⟭_k$. This is achieved by a simple induction on k. $\qquad \square$

Proposition 5.4. $(! N, \alpha)$ *is the final coalgebra of* $N \oslash \_$ *in the category* $\mathcal{W}$.

Proof It suffices to show that if $\sigma : M \to N \oslash M$ is a winning strategy, then $⟬\, \sigma\, ⟭$ is winning.

To see that $⟬\, \sigma\, ⟭$ is total, let $s \in ⟬\, \sigma\, ⟭$ and $so \in P_{!N}$. Then so visits only finite k many copies of N, and so up to retagging it is a play in $M \to (N \oslash \_)^k(M)$, and s a play in $⟬\, \sigma\, ⟭_k$. By totality of $⟬\, \sigma\, ⟭_k$, there is a move p with $sop \in ⟬\, \sigma\, ⟭_k$. Then, up to retagging, sop is also a play in $⟬\, \sigma\, ⟭$.

We next need to check that each infinite play with all even prefixes in $⟬\, \sigma\, ⟭$ is winning. Let s be such an infinite play, with $s|_M$ winning. We must show that $s|_{!N}$ is winning, i.e. $s|_{(N,i)}$ is winning for each i. The infinite play s corresponds to an infinite interaction sequence:

$$M \xrightarrow{\;\sigma\;} N \oslash M \xrightarrow{\;\mathsf{id} \oslash \sigma\;} N \oslash (N \oslash M) \xrightarrow{\;\mathsf{id} \oslash (\mathsf{id} \oslash \sigma)\;} \cdots$$

$$\vdots$$

Then $s|_{(N,i)}$ can also be found in the ith column of the above interaction sequence. By hiding all columns other than the first and the ith, we see a play in $M \to (N \oslash \_)^i(M)$ in $⟬\, \sigma\, ⟭_i$. The first column is $s|_M$ (which is winning), and the ith component of the

second is $s|_{(N,i)}$. Since $(\!(\,\sigma\,)\!)_i$ is a winning strategy, this play is winning, by the winning condition for $\oslash$. $\qquad\square$

Recall that the monoidal unit of a distributive sequoidal category is a terminal object. Thus we may define operations corresponding to dereliction and promotion:

- $\mathsf{der}_N : {!}\,N \to N = \mathsf{unit}_\oslash \circ (\mathsf{id} \oslash \mathsf{t}) \circ \alpha$.

- Given any symmetric comonoid (B, η, δ), and morphism $f : B \to N$, let $f^\dagger : B \to {!}N$ be the (comonoid morphism) $(\!(\,\mathsf{wk} \circ (f \otimes \mathsf{id}) \circ \delta\,)\!)$.

To interpret the contraction rule, we require a further coalgebraic property.

Definition A decomposable, distributive sequoidal category $\mathcal{C}$ has *coalgebraic monoidal exponentials* if:

- For any object A, the endofunctor $A \oslash \_$ has a specified final coalgebra $({!}\,A, \alpha_A)$.

- For any objects A, B, $({!}A{\otimes}{!}B, \alpha_{A,B})$ is a final coalgebra for the endofunctor $(A \times B) \oslash \_$, where $\alpha_{A,B} : {!}A{\otimes}{!}B \to (A \times B) \oslash ({!}A{\otimes}{!}B)$ is the isomorphism:

$$ {!}A{\otimes}{!}B \cong ({!}A{\oslash}{!}B) \times ({!}B{\oslash}{!}A) \cong ((A{\oslash}{!}A){\oslash}{!}B) \times ((B{\oslash}{!}B){\oslash}{!}A) $$
$$ \cong \quad (A \oslash ({!}A{\otimes}{!}B)) \times (B \oslash ({!}A{\otimes}{!}B)) \cong (A \times B) \oslash ({!}A{\otimes}{!}B) $$

The second requirement is equivalent to requiring that the morphism $\langle \mathsf{der}_A \otimes \mathsf{t}, \mathsf{t} \otimes \mathsf{der}_B \rangle^\dagger$ from ${!}A{\otimes}{!}B$ to ${!}(A \times B)$ is an isomorphism. Thus we may define a comonoid $({!}A, \delta : {!}A \to {!}A{\otimes}{!}A, \mathsf{t} : A \to I)$, where δ is the anamorphism of the map $\mathsf{dist}_{A,A,{!}A} \circ \langle \alpha_A, \alpha_A \rangle$.

Proposition 5.5. *If $\mathcal{C}$ has coalgebraic monoidal exponentials then $({!}A, \delta, \mathsf{t})$ is the cofree commutative comonoid on A.*

Proof In other words, the forgetful functor from the category of comonoids on $\mathcal{C}$ into the category $\mathcal{C}$ has a left adjoint which sends A to $({!}A, \delta, \mathsf{t})$. The unit of this adjunction is the dereliction $\mathsf{der}_A : {!}A \to A$: for any $f : B \to A$, $f^\dagger : B \to {!}A$ is the unique comonoid morphism such that $\mathsf{der} \circ f^\dagger$. (Uniqueness follows from finality of ${!}A$.) $\qquad\square$

This cofree commutative comonoid can also be constructed using the technique described in [35]. This approach builds the exponential as a limit of finitary *symmetric tensor powers*, that is, finite tensor products subject to a quotient so that the order that the components are played in is irrelevant. Our use of the asymmetric $\oslash$ enforces a strict left-to-right order, providing a concrete (albeit less generally applicable) alternative to such quotienting.

Proposition 5.6. *The sequoidal closed categories $\mathcal{W}$ and $\mathcal{G}$ are both equipped with coalgebraic monoidal exponentials.*

Proof Follows from Propositions 5.3, 5.4 and the fact $!$ is the cofree commutative comonoid in $\mathcal{G}$ and $\mathcal{W}$ [24]. $\qquad\square$

Definition A *WS!-category* is a distributive, decomposable sequoidal closed category with an object o satisfying linear functional extensionality and coalgebraic monoidal exponentials.

Proposition 5.7. *The categories $(\mathcal{G}, \mathcal{G}_s)$ and $(\mathcal{W}, \mathcal{W}_s)$ enjoy the structure of an WS!-category.*

5.3. Semantics of Rules

We may now describe the interpretation of the rules of our logic (other than those for atoms, quantifiers and equality) in a $\mathsf{WS!}$-category $\mathcal{C}$. Suppose that, for a given context of variables and atoms Φ, we have an interpretation of formulas and sequents over Φ as objects of $\mathcal{C}$, satisfying the following:

$$
\begin{array}{llllll}
[\![\Phi \vdash \mathbf{1}]\!] & = & I & [\![\Phi \vdash \mathbf{0}]\!] & = & I \\
[\![\Phi \vdash \bot]\!] & = & o & [\![\Phi \vdash \top]\!] & = & o \\
[\![\Phi \vdash M \otimes N]\!] & = & [\![\Phi \vdash M]\!] \otimes [\![\Phi \vdash N]\!] & [\![\Phi \vdash P \,\mathbin{⅋}\, Q]\!] & = & [\![\Phi \vdash P]\!] \otimes [\![\Phi \vdash Q]\!] \\
[\![\Phi \vdash M \& N]\!] & = & [\![\Phi \vdash M]\!] \times [\![\Phi \vdash N]\!] & [\![\Phi \vdash P \oplus Q]\!] & = & [\![\Phi \vdash P]\!] \times [\![\Phi \vdash Q]\!] \\
[\![\Phi \vdash M \oslash N]\!] & = & [\![\Phi \vdash M]\!] \oslash [\![\Phi \vdash N]\!] & [\![\Phi \vdash P \lhd Q]\!] & = & [\![\Phi \vdash P]\!] \oslash [\![\Phi \vdash Q]\!] \\
[\![\Phi \vdash M \lhd Q]\!] & = & [\![\Phi \vdash Q]\!] \multimap [\![\Phi \vdash M]\!] & [\![\Phi \vdash P \oslash N]\!] & = & [\![\Phi \vdash N]\!] \multimap [\![\Phi \vdash P]\!] \\
[\![\Phi \vdash {!}N]\!] & = & ![\![\Phi \vdash N]\!] & [\![\Phi \vdash {?}P]\!] & = & ![\![\Phi \vdash P]\!]
\end{array}
$$

$$
\begin{array}{lll}
[\![\Phi \vdash M, \Gamma, N]\!] & = & [\![\Phi \vdash M, \Gamma]\!] \oslash [\![\Phi \vdash N]\!] \\
[\![\Phi \vdash M, \Gamma, P]\!] & = & [\![\Phi \vdash P]\!] \multimap [\![\Phi \vdash M, \Gamma]\!] \\
[\![\Phi \vdash P, \Gamma, N]\!] & = & [\![\Phi \vdash N]\!] \multimap [\![\Phi \vdash P, \Gamma]\!] \\
[\![\Phi \vdash P, \Gamma, Q]\!] & = & [\![\Phi \vdash P, \Gamma]\!] \oslash [\![\Phi \vdash Q]\!]
\end{array}
$$

(For atom and quantifier-free formulas, these equations *define* an interpretation of formulas and sequents in $\mathcal{C}$.) Then we may give an interpretation of each proof rule except those for atoms, quantifiers, and equality as an operation on morphisms in $\mathcal{C}$. These typically involve an operation on the head formula of the sequence "under" a context consisting of its tail, and so we define distributivity maps to allow this:

We define endofunctors $[\![\Gamma]\!]^b$ on $\mathcal{C}_s$ for each context (possibly empty list of formulas) Γ and $b \in \{+, -\}$. below.

$$
\begin{array}{llllll}
[\![\epsilon]\!]^+ & = & \mathsf{id} & [\![\epsilon]\!]^- & = & \mathsf{id} \\
[\![\Gamma, M]\!]^+ & = & [\![M]\!] \multimap [\![\Gamma]\!]^+ & [\![\Gamma, P]\!]^- & = & [\![P]\!] \multimap [\![\Gamma]\!]^- \\
[\![\Gamma, P]\!]^+ & = & [\![\Gamma]\!]^+ \oslash [\![P]\!] & [\![\Gamma, M]\!]^- & = & [\![\Gamma]\!]^- \oslash [\![M]\!]
\end{array}
$$

Proposition 5.8. *For any sequent A, Γ we have $[\![A, \Gamma]\!] = [\![\Gamma]\!]^b([\![A]\!])$ where b is the polarity of A.*

Proof A simple induction on Γ. $\qquad\qquad\square$

Proposition 5.9. *For any context Γ, $[\![\Gamma]\!]^b$ preserves products.*

Proof Using the distributivity of $\times$ over $\oslash$ and $\multimap$, we can construct isomorphisms $\mathsf{dist}_{b,\Gamma} : [\![\Gamma]\!]^b(A \times B) \cong [\![\Gamma]\!]^b(A) \times [\![\Gamma]\!]^b(B)$ and $\mathsf{dist}^0_{b,\Gamma} : [\![\Gamma]\!]^b(I) \cong I$ by induction on Γ.

$$\mathsf{P_1}\ \frac{}{(\mathsf{dist}^0_{-,\Gamma})^{-1} : \llbracket \Vdash \mathbf{1},\Gamma \rrbracket} \qquad\qquad \mathsf{P_\top}\ \frac{}{\mathsf{id}_o : \llbracket \Vdash \top \rrbracket}$$

$$\mathsf{P_\otimes}\ \frac{\sigma : \llbracket \Vdash M,N,\Gamma \rrbracket \qquad \tau : \llbracket \Vdash N,M,\Gamma \rrbracket}{\llbracket \Gamma \rrbracket^-(\mathsf{dec}^{-1}) \circ \mathsf{dist}^{-1}_{-,\Gamma} \circ \langle \sigma,\tau \rangle : \llbracket \Vdash M \otimes N,\Gamma \rrbracket} \qquad \mathsf{P_\&}\ \frac{\sigma : \llbracket \Vdash M,\Gamma \rrbracket \qquad \tau : \llbracket \Vdash N,\Gamma \rrbracket}{\mathsf{dist}^{-1}_{-,\Gamma} \circ \langle \sigma,\tau \rangle : \llbracket \Vdash M\&N,\Gamma \rrbracket}$$

$$\mathsf{P_{\mathbin{⅋}2}}\ \frac{\sigma : \llbracket \Vdash Q,P,\Gamma \rrbracket}{\sigma \circ \llbracket \Gamma \rrbracket^+(\mathsf{wk} \circ \mathsf{sym}) : \llbracket \Vdash P\mathbin{⅋}Q,\Gamma \rrbracket} \qquad\qquad \mathsf{P_{\mathbin{⅋}1}}\ \frac{\sigma : \llbracket \Vdash P,Q,\Gamma \rrbracket}{\sigma \circ \llbracket \Gamma \rrbracket^+(\mathsf{wk}) : \llbracket \Vdash P\mathbin{⅋}Q,\Gamma \rrbracket}$$

$$\mathsf{P_{\oplus 1}}\ \frac{\sigma : \llbracket \Vdash P,\Delta \rrbracket}{\sigma \circ \llbracket \Delta \rrbracket^+(\pi_1) : \llbracket \Vdash P \oplus Q,\Delta \rrbracket} \qquad\qquad \mathsf{P_{\oplus 2}}\ \frac{\sigma : \llbracket \Vdash Q,\Delta \rrbracket}{\sigma \circ \llbracket \Delta \rrbracket^+(\pi_2) : \llbracket \Vdash P \oplus Q,\Delta \rrbracket}$$

$$\mathsf{P^{\mathbin{⅋}}_\perp}\ \frac{\sigma : \llbracket \Vdash \perp,P\mathbin{⅋}Q,\Gamma \rrbracket}{\llbracket \Gamma \rrbracket^-(\mathsf{pasc}_{-\circ} \circ (\mathsf{sym} \multimap \mathsf{id})) \circ \sigma : \llbracket \Vdash \perp,P,Q,\Gamma \rrbracket} \qquad \mathsf{P^+_\perp}\ \frac{\sigma : \llbracket \Vdash P \rrbracket}{\Lambda_I(\sigma) : \llbracket \Vdash \perp,P \rrbracket}$$

$$\mathsf{P^\oslash_\perp}\ \frac{\sigma : \llbracket \Vdash \perp,P \oslash N,\Gamma \rrbracket}{\llbracket \Gamma \rrbracket^-(\mathsf{lfe}^{-1}) \circ \sigma : \llbracket \Vdash \perp,P,N,\Gamma \rrbracket} \qquad\qquad \mathsf{P^-_\top}\ \frac{\sigma : \llbracket \Vdash N \rrbracket}{\mathsf{unit}_{-\circ} \circ (\sigma \multimap \mathsf{id}) : \llbracket \Vdash \top,N \rrbracket}$$

$$\mathsf{P^\otimes_\top}\ \frac{\sigma : \llbracket \Vdash \top,M \otimes N,\Gamma \rrbracket}{\sigma \circ \llbracket \Gamma \rrbracket^+((\mathsf{sym} \multimap \mathsf{id}) \circ \mathsf{pasc}^{-1}_{-\circ}) : \llbracket \Vdash \top,M,N,\Gamma \rrbracket} \qquad \mathsf{P^\triangleleft_\top}\ \frac{\sigma : \llbracket \Vdash \top,N \triangleleft P,\Gamma \rrbracket}{\sigma \circ \llbracket \Gamma \rrbracket^+(\mathsf{lfe}) : \llbracket \Vdash \top,N,P,\Gamma \rrbracket}$$

$$\mathsf{P^-_\perp}\ \frac{\sigma : \llbracket \Vdash \perp,\Gamma \rrbracket}{\llbracket \Gamma \rrbracket^-(\mathsf{abs}^{-1}) \circ \sigma : \llbracket \Vdash \perp,N,\Gamma \rrbracket} \qquad\qquad \mathsf{P_\triangleleft}\ \frac{\sigma : \llbracket \Vdash A,P,\Gamma \rrbracket}{\sigma : \llbracket \Vdash A \triangleleft P,\Gamma \rrbracket}$$

$$\mathsf{P^+_\top}\ \frac{\sigma : \llbracket \top,\Gamma \rrbracket}{\sigma \circ \llbracket \Gamma \rrbracket^+(\mathsf{abs}) : \llbracket \top,P,\Gamma \rrbracket} \qquad\qquad \mathsf{P_\oslash}\ \frac{\sigma : \llbracket \Vdash A,N,\Gamma \rrbracket}{\sigma : \llbracket \Vdash A \oslash N,\Gamma \rrbracket}$$

5.4. Semantics of Proof Rules

Define $\sigma : \llbracket \vdash \Gamma \rrbracket$ if:

- $\Gamma = N,\Gamma'$, and $\sigma : I \to \llbracket \Gamma \rrbracket$ in $\mathcal{C}$.

- $\Gamma = P,\Gamma'$, and $\sigma : \llbracket \Gamma \rrbracket \to o$ in $\mathcal{C}$.

Semantics of the core rules as operations on morphisms are given in Figure 9 and the other rules in Figures 10 and 11. The rules involving the exponential are treated separately in Figure 12. Note that in each case, the interpretation in the WS!-category $\mathcal{W}$ agrees with the informal exposition in Section 3.2.

In the semantics of $\mathsf{P_{cut}}$ we use an additional construction. If $\tau : I \to \llbracket N,\Delta \rrbracket$ define (strict) $\tau^{\circ-}_{M,\Gamma} : \llbracket M,\Gamma,N^\perp \rrbracket \to \llbracket M,\Gamma,\Delta \rrbracket$ to be $\mathsf{unit}_{-\circ} \circ (\tau \multimap \mathsf{id}_{\llbracket M,\Gamma \rrbracket})$ if $|\Delta| = 0$ and $\mathsf{pasc}^n_{-\circ} \circ (\Lambda^{-n}\Lambda_I^{-1}\tau \multimap \mathsf{id}_{\llbracket M,\Gamma \rrbracket})$ if $|\Delta| = n+1$. Define (strict) $\tau^{\circ+}_{P,\Gamma} : \llbracket P,\Gamma,\Delta \rrbracket \to \llbracket P,\Gamma,N^\perp \rrbracket$ to be $(\mathsf{id}_{\llbracket P,\Gamma \rrbracket} \oslash \tau) \circ \mathsf{unit}^{-1}_\oslash$ if $|\Delta| = 0$ and $(\mathsf{id} \oslash \Lambda^{-n}\Lambda_I^{-1}\tau) \circ ((\mathsf{id}_{\llbracket P,\Gamma \rrbracket} \oslash \mathsf{sym}) \circ \mathsf{pasc}^{-1})^n$ if $|\Delta| = n+1$. In some of the rules in Figure 11 we omit some pasc isomorphisms for clarity.

6. Semantics of atoms, quantifiers and equality

We shall now complete the semantics of WS1 by interpreting atoms and quantifiers based on our categories of games and strategies. (The requisite structure could be axiomatised for any WS!-category, but we shall not do so here.) We have seen that a sequent

Figure 10: Categorical Semantics for WS1 (other rules, part 1)

$$\frac{\sigma : [\![\vdash M', \Gamma, M, N, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{psym}) \circ \sigma : [\![\vdash M', \Gamma, N, M, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P, \Gamma, M, N, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{psym}_{\multimap}) : [\![\vdash P, \Gamma, N, M, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash M, \Gamma, P, Q, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{psym}_{\multimap}) \circ \sigma : [\![\vdash M, \Gamma, Q, P, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P', \Gamma, P, Q, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{psym}) : [\![\vdash P', \Gamma, Q, P, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P, \Gamma, M, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+((\mathsf{af} \multimap \mathsf{id}) \circ \mathsf{unit}_{\multimap}^{-1}) : [\![\vdash P, \Gamma, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash N, \Gamma, M, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{unit}_{\oslash} \circ (\mathsf{id} \oslash \mathsf{af})) \circ \sigma : [\![\vdash N, \Gamma, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash M, \Gamma, \Delta]\!]}{[\![\Delta]\!]^-((\mathsf{af} \multimap \mathsf{id}) \circ \mathsf{unit}_{\multimap}^{-1}) \circ \sigma : [\![\vdash M, \Gamma, P, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P, \Gamma, \Delta^+]\!]}{\sigma \circ [\![\Delta]\!]^-(\mathsf{unit}_{\oslash} \circ (\mathsf{id} \oslash \mathsf{af})) : [\![\vdash P, \Gamma, Q, \Delta^+]\!]}$$

$$\frac{\sigma : [\![\vdash N, \Gamma, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{unit}_{\oslash}^{-1}) \circ \sigma : [\![\vdash N, \Gamma, \mathbf{1}, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P, \Gamma, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{unit}_{\multimap}) : [\![\vdash P, \Gamma, \mathbf{1}, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash M, \Gamma, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{unit}_{\multimap}^{-1}) \circ \sigma : [\![\vdash M, \Gamma, \mathbf{0}, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P, \Gamma, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{unit}_{\oslash}) : [\![\vdash P, \Gamma, \mathbf{0}, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash M', \Gamma, M, N, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{pasc}) \circ \sigma : [\![\vdash M', \Gamma, M \otimes N, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P, \Gamma, M, N, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{pasc}_{\multimap}) : [\![\vdash P, \Gamma, M \otimes N, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P', \Gamma, P, Q, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{pasc}^{-1}) : [\![\vdash P', \Gamma, P \,\invamp\, Q, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash M, \Gamma, P, Q, \Delta]\!]}{[\![\Delta]\!]^+(\mathsf{pasc}_{\multimap}^{-1}) \circ \sigma : [\![\vdash M, \Gamma, P \,\invamp\, Q, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash M, \Gamma, P_i, \Delta]\!]}{[\![\Delta]\!]^-(\pi_i \multimap \mathsf{id}) \circ \sigma : [\![\vdash M, \Gamma, P_1 \oplus P_2, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash Q, \Gamma, P_i, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{id} \oslash \pi_i) : [\![\vdash Q, \Gamma, P_1 \oplus P_2, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash N, \Gamma, M_1 \& M_2, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{id} \oslash \pi_i) \circ \sigma : [\![\vdash N, \Gamma, M_i, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash Q, \Gamma, M_1 \& M_2, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\pi_i \multimap \mathsf{id}) : [\![\vdash Q, \Gamma, M_i, \Delta]\!]}$$

$$\mathsf{P_{mul}} \frac{\sigma : [\![\Vdash M, \Gamma, \Delta^+]\!] \qquad \tau : [\![\Vdash N, \Delta_1^+]\!]}{\Lambda_I \Lambda(\mathsf{wk} \circ (\Lambda_I^{-1}(\sigma) \otimes \Lambda_I^{-1}(\tau))) : [\![\Vdash M, \Gamma, N, \Delta^+, \Delta_1^+]\!]}$$

$$\mathsf{P_{cut}} \frac{\sigma : [\![\Vdash M, \Gamma, N^\perp, \Gamma_1]\!] \qquad \tau : [\![\Vdash N, \Delta^+]\!]}{[\![\Gamma_1]\!]^-(\tau_{M,\Gamma}^{\circ-}) \circ \sigma : [\![\Vdash M, \Gamma, \Delta^+, \Gamma_1]\!]}$$

$$\mathsf{P_{cut}} \frac{\sigma : [\![\Vdash P, \Gamma, N^\perp, \Gamma_1]\!] \qquad \tau : [\![\Vdash N, \Delta^+]\!]}{\sigma \circ [\![\Gamma_1]\!]^+(\tau_{P,\Gamma}^{\circ+}) : [\![\Vdash P, \Gamma, \Delta^+, \Gamma_1]\!]}$$

$$\mathsf{P_{id}} \frac{}{\Lambda_I(\mathsf{id}) : [\![\Vdash N, N^\perp]\!]}$$

$$\mathsf{P_{cut}^0} \frac{\sigma : [\![\Vdash N^\perp]\!] \qquad \tau : [\![\Vdash N, Q]\!]}{\sigma \circ \Lambda_I^{-1}(\tau) : [\![\Vdash Q]\!]}$$

$$\mathsf{P_{\multimap}} \frac{\sigma : [\![\Vdash M, \Gamma, P]\!] \qquad \tau : [\![\Vdash N, \Delta^+]\!]}{\mathsf{psym}_\multimap \circ \Lambda_I(\Lambda_I^{-1}(\tau) \multimap \Lambda_I^{-1}(\sigma)) : [\![\Vdash M, \Gamma, P \oslash N, \Delta^+]\!]}$$

$$\mathsf{P_{id\oslash}} \frac{\sigma : [\![\Vdash N, Q, \Delta^+]\!]}{\Lambda_I \Lambda((\mathsf{id} \oslash \Lambda^{-1}\Lambda_I^{-1}(\sigma) \circ \mathsf{sym}) \circ \mathsf{pasc}_\oslash \circ \mathsf{wk} \circ \mathsf{sym}) : [\![\Vdash M, N, M^\perp \lhd Q, \Delta^+]\!]}$$

$X; \Theta \vdash \Gamma$ of WS1 can be interpreted as a family of games, indexed over Θ-satisfying $\mathcal{L}$-models over X. We shall interpret a proof of $X; \Theta \vdash \Gamma$ as a uniform family of strategies for each such game.

For example, the family denoted by $\top \oslash (\phi \lhd \top)$ has games of the following form:
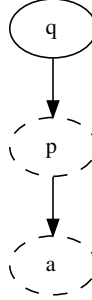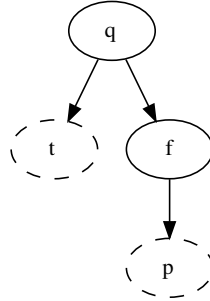

Here we represent the forest of plays P_A directly. The moves in dotted circles are only available if $(L, v) \models \overline{\phi}$. There is a unique total strategy on the (positive) game above in both cases, and this family is uniform in the sense that the strategy on models which satisfy ϕ is a *substrategy* of the strategy on models satisfying $\overline{\phi}$ — if $(L, v) \models \phi$ and $(L', v') \models \overline{\phi}$ then $\sigma_{[\![\top \oslash (\phi \lhd \top)]\!](L,v)} \subseteq \sigma_{[\![\top \oslash (\phi \lhd \top)]\!](L',v')}$.

In contrast, consider the formula $\bot \lhd (\overline{\phi} \oplus (\top \oslash \phi))$. The game forest is given as follows, using the same notation as above:

37

Core rules:

$$\mathsf{P}_! \ \frac{\sigma : [\![\vdash N, !N, \Gamma]\!]}{[\![\Gamma]\!]^-(\alpha^{-1}) \circ \sigma : [\![\vdash !N, \Gamma]\!]} \qquad \mathsf{P}_? \ \frac{\sigma : [\![\vdash P, ?P, \Gamma]\!]}{\sigma \circ [\![\Gamma]\!]^+(\alpha) : [\![\vdash ?P, \Gamma]\!]}$$

Other rules:

$$\mathsf{P}_{\mathsf{ana}} \ \frac{\sigma : [\![\vdash M, P^\perp, P]\!]}{\Lambda_I(\mathbb{C} \ \Lambda_I^{-1}(\sigma) \ \mathbb{D}) : [\![\vdash !M, P]\!]}$$

$$\frac{\sigma : [\![\vdash P, \Gamma, !M, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{der} \multimap \mathsf{id}) : [\![\vdash P, \Gamma, M, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P, \Gamma, !M, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+((\delta \multimap \mathsf{id}) \circ \mathsf{pasc}_{\multimap}^{-1}) : [\![\vdash P, \Gamma, !M, !M, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash N, \Gamma, !M, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{pasc} \circ (\mathsf{id} \oslash \delta)) \circ \sigma : [\![\vdash N, \Gamma, !M, !M, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash M, \Gamma, ?P, ?P, \Delta]\!]}{[\![\Delta]\!]^-((\delta \multimap \mathsf{id}) \circ \mathsf{pasc}_{\multimap}^{-1}) \circ \sigma : [\![\vdash M, \Gamma, ?P, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash Q, \Gamma, ?P, ?P, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{pasc} \circ (\mathsf{id} \oslash \delta)) : [\![\vdash Q, \Gamma, ?P, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash Q, \Gamma, P, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{id} \oslash \mathsf{der}) : [\![\vdash Q, \Gamma, ?P, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash !M, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{der}) \circ \sigma : [\![\vdash M, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash N, \Gamma, !M, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{id} \oslash \mathsf{der}) \circ \sigma : [\![\vdash N, \Gamma, M, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash !M, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{con}) \circ \sigma : [\![\vdash !M, !M, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash ?P, ?P, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{con}) : [\![\vdash ?P, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash P, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{der}) : [\![\vdash ?P, \Delta]\!]}$$

$$\frac{\sigma : [\![\vdash M, \Gamma, P, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{der} \multimap \mathsf{id}) \circ \sigma : [\![\vdash M, \Gamma, ?P, \Delta]\!]}$$

There is a family of strategies on this (negative) game: if ϕ is true, Player plays $\mathtt{f}$ and if $\overline{\phi}$ is true, Player plays $\mathtt{t}$. However, this strategy is not uniform as the choice of second move depends on the truth value of ϕ in the appropriate $\mathcal{L}$-structure. Correspondingly, the formula is not provable in WS1.

We now formalise this notion of uniformity of strategies as a naturality property.

6.1. Uniform Strategies

6.1.1. Game Embeddings

We wish to formalise categorically the notion of a game A being a subgame of B: we can then state that a family of strategies is uniform if whenever A is a subgame of B, the

restriction of σ_B to A is σ_A. If we consider games as trees, we require a tree embedding from P_A into P_B. We use the following machinery:

Definition Let $\mathcal{C}$ be a poset-enriched category. The category $\mathcal{C}_e$ has the same objects as $\mathcal{C}$ and a map $A \to B$ in $\mathcal{C}_e$ consists of a pair (i_f, p_f) where $i_f : A \to B$ and $p_f : B \to A$ in $\mathcal{C}$, such that $p_f \circ i_f = \mathsf{id}$ and $i_f \circ p_f \sqsubseteq \mathsf{id}$.

- The identity is given by $(\mathsf{id}, \mathsf{id})$.

- For composition, set $(i_f, p_f) \circ (i_g, p_g) = (i_f \circ i_g, p_g \circ p_f)$. We need to check this is a valid pairing: $p_{f \circ g} \circ i_{f \circ g} = p_g \circ p_f \circ i_f \circ i_g = p_g \circ \mathsf{id} \circ i_g = \mathsf{id}$ and $i_{f \circ g} \circ p_{f \circ g} = i_f \circ i_g \circ p_g \circ p_f \sqsubseteq i_f \circ \mathsf{id} \circ p_f = i_f \circ p_f \sqsubseteq \mathsf{id}$.

- It is clear that composition is associative and that $f = f \circ \mathsf{id} = \mathsf{id} \circ f$.

Let $\mathcal{G}$ denote the poset-enriched category of games and (not-necessarily winning) strategies, and $\mathcal{G}_s$ its subcategory of strict strategies, with $\sqsubseteq$ given by strategy inclusion. A tree embedding of A into B corresponds to a map $A \to B$ in $\mathcal{G}_e$.

Proposition 6.1. *If $f : A \to B$ in $\mathcal{G}_e$ then i_f and p_f are strict.*

Proof If i_f responds to an opening move in B with a move in B then so does $i_f \circ p_f$ and so $i_f \circ p_f \sqsubseteq \mathsf{id}$ fails. Similarly, if p_f responds to an opening move in A with a move in A then so does $p_f \circ i_f$ and so $p_f \circ i_f = \mathsf{id}$ fails. $\qquad\square$

We can thus define identity-on-objects functors $i : \mathcal{G}_e \to \mathcal{G}_s$ and $p : \mathcal{G}_e \to \mathcal{G}_s^{\mathsf{op}}$. We can show that our operations on games lift to functors on $\mathcal{G}_e$.

Proposition 6.2. *All of the operations $\multimap, \oslash, \otimes, \&, !$ extend to covariant (bi)functors on $\mathcal{G}_e$.*

Proof Each case exploits functoriality and monotonicity of the relevant operation. We just give an example: set $(i, p) \multimap (i', p') = (p \multimap i', i \multimap p')$. Then $(i \multimap p') \circ (p \multimap i') = (p \circ i) \multimap (p' \circ i') = \mathsf{id} \multimap \mathsf{id} = \mathsf{id}$ and $(p \multimap i') \circ (i \multimap p') = (i \circ p) \multimap (i' \circ p') \sqsubseteq \mathsf{id} \multimap \mathsf{id} = \mathsf{id}$. $\qquad\square$

6.1.2. Lax natural Transformations

Given an embedding $e : A \to B$ and strategies $\sigma_A : A$, $\sigma_B : B$, σ_B *restricts to* σ_A if $\sigma_A = p_e \circ \sigma_B$. We generalise this idea using the notion of *lax natural transformations*.

Definition Let $\mathcal{C}$ be a category, $\mathcal{D}$ a poset-enriched category and $F, G : \mathcal{C} \to \mathcal{D}$. A *lax natural transformation* $F \Rightarrow G$ is a family of arrows $\mu_A : F(A) \to G(A)$ such that $\eta_B \circ F(f) \sqsupseteq G(f) \circ \eta_A$.

$$
\begin{array}{ccc}
F(A) & \xrightarrow{\ \mu_A\ } & G(A) \\
{\scriptstyle F(f)}\big\downarrow & \sqsupseteq & \big\downarrow{\scriptstyle G(f)} \\
F(B) & \xrightarrow[\ \mu_B\ ]{} & G(B)
\end{array}
$$

We can compose lax natural transformations using vertical composition. There is also a form of horizontal composition, provided that one of the two functors is the identity: Let $H, G : \mathcal{C} \to \mathcal{D}$ and $\mu : G \Rightarrow H$ a lax natural transformation. Then a) if $F : \mathcal{B} \to \mathcal{C}$ then there is a lax natural transformation $\mu F : G \circ F \Rightarrow H \circ F$ given by $(\mu F)_A = \mu_{F(A)}$ and b) if $J : \mathcal{D} \to \mathcal{E}$ is monotonic then there is a lax natural transformation $J\mu : J \circ G \to J \circ H$ given by $(J\mu)_A = J(\mu_A)$.

6.1.3. Uniform Winning Strategies

Definition Let $F, G : \mathcal{C} \to \mathcal{G}_e$. A *uniform strategy* from F to G is a lax natural transformation $\sigma : i \circ F \Rightarrow i \circ G$. A *uniform total strategy* is a uniform strategy σ where each σ_A is total. A *uniform winning strategy* is a uniform strategy where each σ_A is winning.

If $f : A \to B$, the lax naturality condition is that $i_{G(f)} \circ \sigma_A \sqsubseteq \sigma_B \circ i_{F(f)}$. Thus $\sigma_A = p_{G(f)} \circ i_{G(f)} \circ \sigma_A \sqsubseteq p_{G(f)} \circ \sigma_B \circ i_{F(f)}$. But since σ_A is total, it is maximal in the ordering $\sqsubseteq$ and we must have $\sigma_A = p_{G(f)} \circ \sigma_B \circ i_{F(f)}$. Similarly, we see that $\sigma_A = p_{G(f)} \circ \sigma_B \circ i_{F(f)}$ implies the lax naturality condition as $i_{G(f)} \circ \sigma_A = i_{G(f)} \circ p_{G(f)} \circ \sigma_B \circ i_{F(f)} \sqsubseteq \sigma_B \circ i_{F(f)}$. Thus, lax naturality captures the fact that σ_A is determined by σ_B via restriction. If F is the constant functor κ_I, this reduces to $\sigma_A = p_{G(e)} \circ \sigma_B$.

We can construct a WS-category of uniform strategies over a base category $\mathcal{C}$. Let $\mathcal{G}^{\mathcal{C}}$ be the category where:

- Objects are functors $\mathcal{C} \to \mathcal{G}_e$

- An arrow $F \to G$ is a uniform strategy $F \Rightarrow G$

- Composition is given by vertical composition of lax natural transformations

- The identity on a functor F is given by the lax natural transformation $\eta : F \Rightarrow F$ where $\eta_A = \mathsf{id}_{F(A)}$. It is clear that this is lax natural.

Similarly, we can construct a category $\mathcal{W}^{\mathcal{C}}$ of functors and uniform winning strategies.

Proposition 6.3. *$\mathcal{G}^{\mathcal{C}}$ is a WS!-category.*

Proof We first exhibit the symmetric monoidal structure. $F \otimes G$ is defined to be $\otimes \circ (F \times G) \circ \Delta$ where $\Delta : \mathcal{C} \to \mathcal{C} \times \mathcal{C}$ is the diagonal. So, $(F \otimes G)(A) = F(A) \otimes G(A)$. On arrows, we set $(\eta \otimes \rho)_A = \eta_A \otimes \rho_A$. We need to show that if $f : L \to K$ then $(i_{A(f)} \otimes i_{C(f)}) \circ (\eta_K \otimes \rho_K) \sqsupseteq (i_{B(f)} \otimes i_{D(f)}) \circ (\eta_L \otimes \rho_L)$. That is, we need to show that $(i_{A(f)} \circ \eta_K) \otimes (i_{C(f)} \circ \rho_K) \sqsupseteq (i_{B(f)} \circ \eta_L) \otimes (i_{D(f)} \circ \rho_L)$. But this is clear by lax naturality of η and ρ and monotonicity of $\otimes$.

The tensor unit I is the constant functor, sending all objects to the game I and arrows to id_I.

The morphisms assoc, $\mathsf{runit}_\otimes$, $\mathsf{lunit}_\otimes$ and sym are defined pointwise: for example, $(\mathsf{assoc}_{F,G,H})_X = \mathsf{assoc}_{F(X),G(X),H(X)}$. To check for lax naturality, we must use horizontal composition. For example, consider the map $\mathsf{assoc} : (F \otimes G) \otimes H \to F \otimes (G \otimes H)$ defined pointwise as described. The domain is $(F \otimes G) \otimes H = ((\_ \otimes \_) \otimes \_) \circ (i \circ F \times i \circ G \times i \circ H) \circ \Delta_3$ where Δ_3 is the diagonal functor $\mathcal{C} \to \mathcal{C} \times \mathcal{C} \times \mathcal{C}$. Similarly, the codomain is $(\_ \otimes (\_ \otimes \_)) \circ (i \circ F \times i \circ G \times i \circ H) \circ \Delta_3$. We can thus see that assoc is equal to the

horizontal composition $\mathsf{assoc} J$ where $J = (i \circ F \times i \circ G \times i \circ H) \circ \Delta_3$ and assoc is the natural transformation $\_ \otimes (\_ \otimes \_) \Rightarrow (\_ \otimes \_) \otimes \_$ in $\mathcal{G}_s$.

$$
\begin{array}{ccccccc}
\mathcal{C} & \xrightarrow{\Delta_3} & \mathcal{C} \times \mathcal{C} \times \mathcal{C} & \xrightarrow{i \circ F \times i \circ G \times i \circ H} & \mathcal{G}_s \times \mathcal{G}_s \times \mathcal{G}_s & \xrightarrow{\_ \otimes (\_ \otimes \_)} & \mathcal{G}_s \\
{\scriptstyle \mathsf{id}} \Big\Downarrow & & {\scriptstyle \mathsf{id}} \Big\Downarrow & & & {\scriptstyle \mathsf{assoc}} \Big\Downarrow & \\
\mathcal{C} & \xrightarrow{\Delta_3} & \mathcal{C} \times \mathcal{C} \times \mathcal{C} & \xrightarrow{i \circ F \times i \circ G \times i \circ H} & \mathcal{G}_s \times \mathcal{G}_s \times \mathcal{G}_s & \xrightarrow{(\_ \otimes \_) \otimes \_} & \mathcal{G}_s
\end{array}
$$

One can similarly express the other monoidal isomorphisms in this way to see lax naturality. The coherence equations of symmetric monoidal categories inherit pointwise from $\mathcal{G}$.

Symmetric monoidal closure, products, sequoidal closure and linear functional extensionality lift pointwise from $\mathcal{G}$ using horizontal composition. We can also show that the coalgebraic monoidal exponential structure lifts from $\mathcal{G}$. □

Proposition 6.4. $\mathcal{W}^{\mathcal{C}}$ *is a WS!-category.*

Proof We proceed precisely as in Proposition 6.3, lifting the structure of a WS!-category in $\mathcal{W}$ to that in $\mathcal{W}^{\mathcal{C}}$. In particular, pointwise-winningness of the relevant morphisms in $\mathcal{W}^{\mathcal{C}}$ inherits from the winningness in $\mathcal{W}$. □

6.2. Quantifiers

6.2.1. Category of $\mathcal{L}$-structures

Definition Given a set of variables X and set of atomic formulas Θ, we let $\mathcal{M}_{\Theta}^{X}$ denote the category of Θ-satisfying $\mathcal{L}$-models over X. Objects are $\mathcal{L}$-models over X that satisfy each formula in Θ. A morphism $(L, v) \to (L', v')$ is a map $f : |L| \to |L'|$ such that:

- For each $x \in X$, $v'(x) = f(v(x))$

- If $(L, v) \models \overline{\phi}(\overrightarrow{a})$ for $\overrightarrow{a} \in |L|^{\mathsf{ar}(\phi)}$ then $(L', v') \models \overline{\phi}(\overrightarrow{f(a)})$

- For each function symbol g in $\mathcal{L}$, $f(I_L(g)(\overrightarrow{a})) = I_{L'}(g)(\overrightarrow{f(a)})$.

Note that since the positive atoms include inequality, such morphisms must be injective. Also note that if $f : (L, v) \to (L', v')$ and $(L, v) \models \overline{\phi}(\overrightarrow{s})$ then $(L', v') \models \overline{\phi}(\overrightarrow{s})$.

If v is a valuation on X, define $v[x \mapsto l]$ on $X \cup \{x\}$ to be the valuation sending y to $v(y)$ if $y \neq x$, and x to l. Given $f : (L, v) \to (M, w)$ in $\mathcal{M}_X^{\Theta}$ and s a term with $FV(s) \subseteq X$, f is also a map $(L, v[x \mapsto v(s)]) \to (M, w[x \mapsto w(s)])$ in $\mathcal{M}_{X \cup \{x\}}^{\Theta}$. We know that f preserves all of the valuations other than x, and for x we see that $f(v[x \mapsto v(s)](x)) = f(v(s)) = w(s) = w[x \mapsto w(s)](x)$.

We will give semantics of sequents $X; \Theta \vdash \Gamma$ as functors $\mathcal{M}_{\Theta}^{X} \to \mathcal{G}_e$, and proofs as uniform winning strategies.

6.2.2. Quantifiers as Adjoints

In this section, we will describe an adjunction that will allow us to interpret the quantifiers.

- If $FV(s) \subseteq X$ we can define a functor $\mathsf{set}_s^x : \mathcal{M}_X^\Theta \to \mathcal{M}_{X \uplus \{x\}}^\Theta$ by $\mathsf{set}_s^x(L, v) = (L, v[x \mapsto v(s)])$ and if $f : (L, v) \to (M, w)$ we set $\mathsf{set}_s^x(f) = f$. We need to check that $\mathsf{set}_s^x(f)$ is a valid morphism. We know that $\mathsf{set}_s^x(f)$ preserves all variables in X, and $\mathsf{set}_s^x(f)(v[x \mapsto v(s)](x)) = f(v(s)) = w(s) = w[x \mapsto w(s)](x)$ as required. It is clear that set_s^x is functorial.

  From this we can extract a functor $\mathsf{set}_s'^x : \mathcal{W}^{\mathcal{M}_{X \uplus \{x\}}^\Theta} \to \mathcal{W}^{\mathcal{M}_X^\Theta}$, mapping F to $F \circ \mathsf{set}_s^x$, with an action on arrows defined by horizontal composition.

- Provided x does not occur in Θ, there is an evident forgetful functor $U_x : \mathcal{M}_{X \uplus \{x\}}^\Theta \to \mathcal{M}_X^\Theta$ mapping (L, v) to $(L, v - x)$. From this we can extract a functor $U_x' : \mathcal{W}^{\mathcal{M}_X^\Theta} \to \mathcal{W}^{\mathcal{M}_{X \uplus \{x\}}^\Theta}$ mapping F to $F \circ U_x$, with an action on arrows defined by horizontal composition. Note that $U_x \circ \mathsf{set}_s^x = \mathsf{id}$ and so $\mathsf{set}_s'^x \circ U_x' = \mathsf{id}$.

We will show that U_x' has a right adjoint $\forall x.\_$. Assuming empty Γ, this allows us to interpret the rules $\mathsf{P}_\forall$ and $\mathsf{P}_\exists$.

Definition Let $\mathcal{C}$ be a category. We define the category $\mathsf{FamInj}(\mathcal{C})$. An object is a set I and a family of $\mathcal{C}$-objects $\{A_i : i \in I\}$. An arrow $\{A_i : i \in I\} \to \{B_j : j \in J\}$ is a pair $(f, \{f_i : i \in I\})$ where f is an injective function $I \to J$ and each $f_i : A_i \to B_{f(i)}$. We will often write such a map as $(f, \{f_i\})$ when we wish to leave the indexing set implicit.

- Composition is defined by $(f, \{f_i\}) \circ (g, \{g_i\}) = (f \circ g, \{f_{g(i)} \circ g_i\})$.

- The identity $\{A_i : i \in I\} \to \{A_i : i \in I\}$ is given by $(\mathsf{id}, \{\mathsf{id}_{A_i}\})$.

- Satisfaction of the categorical axioms is inherited from $\mathcal{C}$.

Definition Let $F : \mathcal{C} \to \mathcal{D}$. We define $\mathsf{FamInj}(F) : \mathsf{FamInj}(\mathcal{C}) \to \mathsf{FamInj}(\mathcal{D})$. On objects, $\mathsf{FamInj}(F)(\{A_i : i \in I\}) = \{F(A_i) : i \in I\}$. On arrows, we set $\mathsf{FamInj}(F)(f, \{f_i\}) = (f, \{F(f_i)\})$.

We define a distributivity functor $\mathsf{dst} : \mathsf{FamInj}(\mathcal{C}) \times \mathcal{D} \to \mathsf{FamInj}(\mathcal{C} \times \mathcal{D})$ by $\mathsf{dst}(\{A_i : i \in I\}, B) = \{(A_i, B) : i \in I\}$ and $\mathsf{dst}((f, \{f_i\}), g) = (f, \{(f_i, g)\})$.

Suppose F is an object in $\mathcal{W}^{\mathcal{M}_{X \uplus \{x\}}^\Theta}$ (a functor $\mathcal{M}_{X \uplus \{x\}}^\Theta \to \mathcal{G}_e$). We define $\forall x.F$ as an object in $\mathcal{W}^{\mathcal{M}_X^\Theta}$ (a functor $\mathcal{M}_X^\Theta \to \mathcal{G}_e$). We first define a product functor $\mathsf{prod} : \mathsf{FamInj}(\mathcal{G}_e) \to \mathcal{G}_e$. On objects, prod sends $\{G_i : i \in I\}$ to $\prod_{i \in I} G_i$. On arrows, let $f : \{G_j : j \in J\} \to \{H_h : h \in H\}$. The embedding part of $\mathsf{prod}(f)$ is given by $\langle g_h \rangle_h$ where $g_h = i_{f_j} \circ \pi_j$ if $h = f(j)$ and ϵ otherwise. The projection part is given by $\langle p_{f_j} \circ \pi_{f(j)} \rangle_j$. We can check that prod defines a functor into $\mathcal{G}_e$. Finally, given $F : \mathcal{M}_{X \uplus \{x\}}^\Theta \to \mathcal{G}_e$ we define $\forall x.F : \mathcal{M}_X^\Theta \to \mathcal{G}_e$ to be $\mathsf{prod} \circ \mathsf{FamInj}(F) \circ \mathsf{add}_x$.

Proposition 6.5. *The functor* $U_x' : \mathcal{W}^{\mathcal{M}_X^\Theta} \to \mathcal{W}^{\mathcal{M}_{X \uplus \{x\}}^\Theta}$ *has a right adjoint given by* $\forall x.\_ = \mathsf{prod} \circ \mathsf{FamInj}(\_) \circ \mathsf{add}_x : \mathcal{W}^{\mathcal{M}_{X \uplus \{x\}}^\Theta} \to \mathcal{W}^{\mathcal{M}_X^\Theta}$

Proof We must first give the unit of this adjunction. For each F, we must give a uniform winning strategy $\eta : U'_x(\forall x.F) \Rightarrow F$. Such an η is a winning uniform strategy $\mathsf{prod} \circ \mathsf{FamInj}(F) \circ \mathsf{add}_x \circ U_x \Rightarrow F$. Note that $(\mathsf{prod} \circ \mathsf{FamInj}(F) \circ \mathsf{add}_x \circ U_x)(L, v) = \mathsf{prod}(\{F(L, v - x[x \mapsto l]) : l \in L\}) = \prod_{l \in L} F(L, v[x \mapsto l])$. Thus $\eta_{(L,v)}$ must be a winning strategy $\prod_{l \in L} F(L, v[x \mapsto l]) \to F(L, v)$ and we take $\eta_{(L,v)} = \pi_{v(x)}$. One can check that this transformation is lax natural.

Given $f : U'_x(F) \to G$ we must show that there is a unique $\hat{f} : F \to \forall x.G$ such that $f = \eta_G \circ U'_x(\hat{f})$. Let f be such a uniform winning strategy. Then we must give winning strategies $\hat{f}_{(L,v)} : F(L, v) \to \prod_{l \in L} G(L, v[x \mapsto l])$. Set $\hat{f}_{(L,v)} = \langle h_l \rangle_l$ where $h_l : F(L, v) \to G(L, v[x \mapsto l])$ is defined by $f_{(L, v[x \mapsto l])}$. We can check that $\hat{f}$ satisfies lax naturality.

We next need to show that $\hat{f}$ satisfies the universal property. Firstly, we must show that $f = \eta_G \circ U'_x(\hat{f})$. It suffices to show that for each (L, v), $f_{(L,v)} = ((\eta_G) \circ U'_x(\hat{f}))_{L,v}$. Composition in is given by vertical composition. Thus, the RHS is given by $\pi_{v(x)} \circ \langle f_{(L, v[x \mapsto l])} \rangle_l = f_{(L, v[x \mapsto v(x)])} = f_{(L,v)}$ as required.

We need to show that $\hat{f} : F \to \forall x.G$ is the unique uniform strategy satisfying $f = \eta_G \circ U'_x(\hat{f})$. Suppose $h : F \to \forall x.G$ in $\mathcal{W}^{\mathcal{M}^\Theta_X}$ satisfies this property. Then given (L, v) in $\mathcal{M}^\Theta_{X \uplus \{x\}}$, we know that $f_{(L,v)} = \eta_{G(L,v)} \circ h_{(L, v-x)} = \pi_{v(x)} \circ h_{(L, v-x)}$. Let $(L, v) \in \mathcal{M}^\Theta_X$. We must show that $h_{(L,v)} = \hat{f}_{(L,v)} = \langle f_{(L, v[x \mapsto l])} \rangle_l$. Thus we need to show that for each l, $\pi_l \circ h_{(L,v)} = f_{(L, v[x \mapsto l])}$. But consider the model $(L, v[x \mapsto l])$. This is $f_{(L, v[x \mapsto l])} = \pi_{v[x \mapsto l](x)} \circ h_{(L, v[x \mapsto l] - x)} = \pi_l \circ h_{(L,v)}$, as required. $\square$

If $N : \mathcal{M}^\Theta_{X \uplus \{x\}} \to \mathcal{G}_e$ then on objects $[\![\forall x.N]\!](L, v) = \prod_{l \in |L|} [\![N]\!](L, v[x \mapsto l])$. For the action of $\forall x.N$ on arrows, suppose $f : (L, v) \to (L', w)$. Then $[\![\forall x.N]\!](f) : \prod_{l \in |L|} [\![N]\!](L, v[x \mapsto l]) \to \prod_{l \in |L'|} [\![N]\!](L', w[x \mapsto l])$ is given as follows: The embedding part (left to right) is given by $\langle g_m \rangle_m$ where $g_m = \epsilon$ if m is not in the image of f, and $g_m = i[\![N]\!](f) \circ \pi_l$ if $m = f(l)$ (note in this case l is unique by injectivity of f). The projection part is given by $\langle p[\![N]\!](f) \circ \pi_{f(l)} \rangle_l$.

Consider the map $\mathsf{set}'^x_s(\eta) : \forall x.F = \mathsf{set}'^x_s(U'_x(\forall x.F)) \to \mathsf{set}'^x_s(F)$ in the category $\mathcal{M}^\Theta_X$. Pointwise, $\mathsf{set}'^x_s(\eta)_{(L,v)} : \prod_{l \in L} F(L, v[x \mapsto l]) \to F(L, v[x \mapsto v(s)])$ is given by $\pi_{v(s)}$, and so we will write π_s for this map.

6.3. Semantics of Sequents

We define the semantics of sequents $X; \Theta \vdash \Gamma$ as functors $\mathcal{M}^\Theta_X \to \mathcal{G}_e$ inductively, via the equations given in the previous section, extended with the following interpretations of atoms and quantifiers:

$$
\begin{array}{llllll}
[\![\Phi \vdash \phi(\overrightarrow{s})]\!](L, v) & = & I \text{ if } (L, v) \models \overline{\phi}(\overrightarrow{s}) & [\![\Phi \vdash \overline{\phi}(\overrightarrow{s})]\!](L, v) & = & I \text{ if } (L, v) \models \overline{\phi}(\overrightarrow{s}) \\
[\![\Phi \vdash \phi(\overrightarrow{s})]\!](L, v) & = & o \text{ if } (L, v) \models \phi(\overrightarrow{s}) & [\![\Phi \vdash \overline{\phi}(\overrightarrow{s})]\!](L, v) & = & o \text{ if } (L, v) \models \phi(\overrightarrow{s})
\end{array}
$$

$$
\begin{array}{lll}
[\![X; \Theta \vdash \forall x.N]\!] & = & \forall x.[\![X \uplus \{x\}; \Theta \vdash N]\!] \\
[\![X; \Theta \vdash \exists x.P]\!] & = & \forall x.[\![X \uplus \{x\}; \Theta \vdash P]\!]
\end{array}
$$

In the case of atoms, the functors are specified pointwise on objects, and we must also define the (functorial) action on arrows. Let $f : (L, v) \to (L', v')$. If the truth value of $\phi(\overrightarrow{s})$ is the same in (L, v) and (L', v), we use the identity embedding $(\mathsf{id}, \mathsf{id})$. If the

43

truth value of $\phi(\overrightarrow{s})$ is different, we must have $(L, v) \models \phi(\overrightarrow{s})$ and $(L', v) \models \overline{\phi}(\overrightarrow{s})$ since morphisms in $\mathcal{M}_X$ preserve truth of positive atoms. Thus we need an embedding $I \to o$. We can take $(\epsilon_{I \multimap o}, \epsilon_{o \multimap I})$ where ϵ_A is the strategy containing just the empty sequence. Note that $\epsilon_{o \multimap I} \circ \epsilon_{I \multimap o} = \epsilon_I = \mathsf{id}_I$ and $\epsilon_{I \multimap o} \circ \epsilon_{o \multimap I} = \epsilon \sqsubseteq \mathsf{id}_o$ (ϵ is the bottom element with respect to $\sqsubseteq$).

We must check functoriality. We have already noted that if the truth value of $\phi(\overrightarrow{s})$ is the same in (L, v) and (L', v') then $[\![\phi(\overrightarrow{s})]\!](f) = \mathsf{id}$, so in particular $[\![\phi(\overrightarrow{s})]\!](\mathsf{id}) = \mathsf{id}$. For composition, suppose $f : (L, v) \to (L', v')$ and $g : (L', v') \to (L'', v'')$. We can consider the truth value of $\phi(\overrightarrow{s})$ in each of these models (only some cases are possible, as morphisms preserve truth of positive atoms).

$(L, v) \models$	$(L', v') \models$	$(L'', v'') \models$	$[\![\phi(\overrightarrow{s})]\!](g) \circ [\![\phi(\overrightarrow{s})]\!](g) = [\![\phi(\overrightarrow{s})]\!](g \circ f)$
$\phi(\overrightarrow{s})$	$\phi(\overrightarrow{s})$	$\phi(\overrightarrow{s})$	$(\mathsf{id}, \mathsf{id}) \circ (\mathsf{id}, \mathsf{id}) = (\mathsf{id}, \mathsf{id})$
$\phi(\overrightarrow{s})$	$\phi(\overrightarrow{s})$	$\overline{\phi}(\overrightarrow{s})$	$(\epsilon, \epsilon) \circ (\mathsf{id}, \mathsf{id}) = (\epsilon, \epsilon)$
$\phi(\overrightarrow{s})$	$\overline{\phi}(\overrightarrow{s})$	$\overline{\phi}(\overrightarrow{s})$	$(\mathsf{id}, \mathsf{id}) \circ (\epsilon, \epsilon) = (\epsilon, \epsilon)$
$\overline{\phi}(\overrightarrow{s})$	$\overline{\phi}(\overrightarrow{s})$	$\overline{\phi}(\overrightarrow{s})$	$(\mathsf{id}, \mathsf{id}) \circ (\mathsf{id}, \mathsf{id}) = (\mathsf{id}, \mathsf{id})$

6.4. Semantics of Proofs

We now extend the semantics of proof rules given in the previous section with interpretations for the rules for quantifiers, atoms and equality, completing the semantics of WS1.

We first show that if $x \notin FV(\Gamma)$ there is an isomorphism $\mathsf{dist}_\Gamma : [\![\forall x.A, \Gamma]\!] \cong \forall x.[\![A, \Gamma]\!]$ in $\mathcal{W}^{\mathcal{M}_X^\Theta}$. Observe that there is a natural isomorphism

$$\mathsf{dist}_\oslash : {}_{-} \oslash {}_{-} \circ (\mathsf{prod} \times \mathsf{id}) \Rightarrow \mathsf{prod} \circ \mathsf{FamInj}({}_{-} \oslash {}_{-}) \circ \mathsf{dst} : \mathsf{FamInj}(\mathcal{G}_s) \times \mathcal{G}_s \to \mathcal{G}_s$$

which is concretely a family of winning strategies

$$\mathsf{prod}(\{G_i : i \in I\}) \oslash M \to \mathsf{prod}(\{G_i \oslash M : i \in I\})$$

given by $\mathsf{dist}_\oslash = \langle \pi_i \oslash \mathsf{id} \rangle_i$. Each $\mathsf{dist}_\oslash$ is a natural isomorphism in $\mathcal{W}_s$.

Similarly, we can define a natural isomorphism

$$\mathsf{dist}_{\multimap} : \mathsf{prod}(\{M \multimap G_i : i \in I\}) \cong M \multimap \mathsf{prod}(\{G_i : i \in I\})$$

between functors

$${}_{-} \multimap {}_{-} \circ (\mathsf{prod} \times \mathsf{id}) \Rightarrow \mathsf{prod} \circ \mathsf{FamInj}({}_{-} \multimap {}_{-}) \circ \mathsf{dst} : \mathsf{FamInj}(\mathcal{G}_s) \times \mathcal{G}_s \to \mathcal{G}_s.$$

For each Γ, we can then construct a map

$$\mathsf{dist}_{b,\Gamma} : [\![\Gamma]\!]_1^b \circ (\mathsf{prod} \times \mathsf{id}) \cong \mathsf{prod} \circ \mathsf{FamInj}([\![\Gamma]\!]_1^b) \circ \mathsf{dst} : \mathsf{FamInj}(\mathcal{G}_s) \times \mathcal{M}_X^\Theta \to \mathcal{G}_s$$

proceeding by induction on Γ.

Finally, given a sequent A, Γ we define dist_Γ as the following horizontal composition, where b is the polarity of A. It is easy to see by checking pointwise that the functor $\forall x.[\![A, \Gamma]\!]$ is equal to the given decomposition.

$$
\begin{array}{ccccccc}
\forall x.[\![A, \Gamma]\!] : \mathcal{M}_X^\Theta & \xrightarrow{\langle \mathsf{add}_x, \mathsf{id} \rangle} & \mathsf{FamInj}(\mathcal{M}_{X \uplus \{x\}}^\Theta) \times \mathcal{M}_X^\Theta & \xrightarrow{\mathsf{FamInj}(A) \times \mathsf{id}} & \mathsf{FamInj}(\mathcal{G}_s) \times \mathcal{M}_X^\Theta & \xrightarrow{\mathsf{prod} \circ \mathsf{FamInj}([\![\Gamma]\!]_1^b) \circ \mathsf{dst}} & \mathcal{G}_s \\
& \Big\| \mathsf{id} & & \Big\| \mathsf{id} & & \Big\| \mathsf{dist}_{b,\Gamma}^{-1} & \\
[\![\forall x.A, \Gamma]\!] : \mathcal{M}_X^\Theta & \xrightarrow{\langle \mathsf{add}_x, \mathsf{id} \rangle} & \mathsf{FamInj}(\mathcal{M}_{X \uplus \{x\}}^\Theta) \times \mathcal{M}_X^\Theta & \xrightarrow{\mathsf{FamInj}(A) \times \mathsf{id}} & \mathsf{FamInj}(\mathcal{G}_s) \times \mathcal{M}_X^\Theta & \xrightarrow{[\![\Gamma]\!]_1^b \circ (\mathsf{prod} \times \mathsf{id})} & \mathcal{G}_s
\end{array}
$$

Since dist_Γ is a natural isomorphism, and pointwise winning, it is an isomorphism in $\mathcal{W}^{\mathcal{M}_X^\Theta}$.

Proposition 6.6. $\pi_{v(x)} \circ \mathsf{dist}_{\Gamma(L,v)} = [\![\Gamma]\!]^b(\pi_{v(x)})$

Proof We can check this by induction on Γ, as in Proposition 5.9. $\qquad\square$

We next give semantics to the rules involving atoms and quantifiers. We first introduce some notation. Suppose $\mathcal{C}$ is the coproduct of two categories $\mathcal{D}$ and $\mathcal{E}$ (the disjoint union of the two categories, where there are no maps between them). If $F : \mathcal{C} \to \mathcal{G}_e$ we write $F|_\mathcal{D}$ and $F|_\mathcal{E}$ for the restriction of F to $\mathcal{D}$ and $\mathcal{E}$ respectively. If $\eta : F \Rightarrow G$ then we can restrict η to a natural transformation $F|_\mathcal{D} \Rightarrow G|_\mathcal{D}$, and we write $\eta|_\mathcal{D}$ for this restriction. If $\eta : F|_\mathcal{D} \Rightarrow G|_\mathcal{D}$ and $\sigma : F|_\mathcal{E} \Rightarrow G|_\mathcal{E}$ then we write $[\eta, \sigma]_{\mathcal{D},\mathcal{E}}$ for the lax natural transformation defined by $[\eta, \sigma]_A = \eta_A$ if $A \in \mathcal{D}$ and $[\eta, \sigma]_A = \sigma_A$ if $A \in \mathcal{E}$. Lax naturality of $[\eta, \sigma]$ inherits from lax naturality of η and σ, since there are no maps between $\mathcal{D}$ and $\mathcal{E}$ when viewed as subcategories of $\mathcal{C}$. If $\mathcal{C} = \mathcal{M}_X^\Theta$ then we will write $[\eta, \sigma]_{\alpha,\beta}$ for $[\eta, \sigma]_{\mathcal{M}_X^{\Theta,\alpha}, \mathcal{M}_X^{\Theta,\beta}}$.

We construct an isomorphism

$$H_{x,y,z} : \mathcal{M}_X^{\Theta, x=y} \cong \mathcal{M}_{X/\{x,y\} \uplus \{z\}}^{\Theta[\frac{z}{x}, \frac{z}{y}]} : H_{x,y,z}^{-1}$$

with $H_{x,y,z}(M,v) = (M, v[z \mapsto v(x)] - x - y)$ and $H^{-1}(M,v) = (M, v[x \mapsto v(z), y \mapsto v(z)] - z)$. We can show that $[\![(X; \Theta \vdash \Gamma)[\frac{z}{x}, \frac{z}{y}]]\!] = [\![X; \Theta, x = y \vdash \Gamma]\!]H_{x,y,z}^{-1}$ by induction on Γ.

Semantics of the rules involving atoms and quantifiers are given in Figure 13. We must justify lax naturality of $\mathsf{P_{at-}}$: the following diagram must lax commute:

$$
\begin{array}{ccc}
I & \xrightarrow{\;[\![\mathsf{P_{at-}}(p)]\!](M,w)\;} & [\![\phi(\vec{s}), \Gamma]\!](M,w) \\[2mm]
{\scriptstyle \mathsf{id}}\Big\downarrow & \sqsupseteq & \Big\downarrow {\scriptstyle i[\![\phi(\vec{s}), \Gamma]\!](f)} \\[2mm]
I & \xrightarrow[\;[\![\mathsf{P_{at-}}(p)]\!](L,v)\;]{} & [\![\phi(\vec{s}), \Gamma]\!](L,v)
\end{array}
$$

To see this, note that if (L,v) and (M,w) agree on $\phi(\vec{x})$ then the diagram lax commutes by lax naturality of ϵ or $[\![p]\!]$. If they disagree, then we must have $(L,v) \models \overline{\phi(\vec{x})}$ and $(M,w) \models \phi(\vec{x})$. We need to show that $[\![\mathsf{P_{at-}}(p)]\!](L,v) \sqsupseteq i[\![\phi(\vec{x}), \Gamma]\!](f) \circ [\![\mathsf{P_{at-}}(p)]\!](M,w)$. But $[\![\mathsf{P_{at-}}(p)]\!](M,w) = p[\![\phi(\vec{x}), \Gamma]\!](f) \circ [\![\mathsf{P_{at-}}(p)]\!](L,v)$ as both sides map into the terminal object, so $[\![\mathsf{P_{at-}}(p)]\!](L,v) \sqsupseteq i[\![\phi(\vec{x}), \Gamma]\!](f) \circ p[\![\phi(\vec{x}), \Gamma]\!](f) \circ [\![\mathsf{P_{at-}}(p)]\!](L,v) = i[\![\phi(\vec{x}), \Gamma]\!](f) \circ [\![\mathsf{P_{at-}}(p)]\!](M,w)$.

7. Full Completeness

We next show a full completeness result for the *function-free* fragment of $\mathsf{WS1}$: in this section we assume that $\mathcal{L}$ contains no function symbols. Thus, the only uses of the $\mathsf{P_\exists}$ rule are of the form $\mathsf{P_\exists^y}$ where y is some variable in scope.

$$\mathsf{P_{at-}} \quad \frac{\sigma : [\![\Theta, \overline{\phi}(\overrightarrow{s}) \vdash \bot, \Gamma]\!]}{[\sigma, \epsilon]_{\overline{\phi}(\overrightarrow{s}), \phi(\overrightarrow{s})} : [\![\Theta \vdash \phi(\overrightarrow{s}), \Gamma]\!]}$$

$$\mathsf{P_{at+}} \quad \frac{\sigma : [\![\Theta, \overline{\phi}(\overrightarrow{s}) \vdash \top, \Gamma]\!]}{\sigma : [\![\Theta, \overline{\phi}(\overrightarrow{s}) \vdash \overline{\phi}(\overrightarrow{x}), \Gamma]\!]}$$

$$\mathsf{P_{ma}^{x,y,z}} \quad \frac{\sigma : [\![(X; \Theta \vdash \Gamma)[\tfrac{z}{x}, \tfrac{z}{y}]]\!] \qquad \tau : [\![X; \Theta, x \neq y \vdash \Gamma]\!]}{[\sigma H_{x,y,z}, \tau]_{x=y, x \neq y} : [\![X; \Theta \vdash \Gamma]\!]}$$

$$\mathsf{P_{\neq}} \quad \frac{}{\emptyset : [\![\Theta, x \neq x \vdash \Gamma]\!]}$$

$$\mathsf{P_{\forall}} \quad \frac{\sigma : [\![X \uplus \{x\}; \Theta \vdash N, \Gamma]\!]}{\mathsf{dist}_\Gamma^{-1} \circ \hat{\sigma} : [\![X; \Theta \vdash \forall x.N, \Gamma]\!]} \quad x \notin FV(\Theta, \Gamma)$$

$$\mathsf{P_{\exists}^{s}} \quad \frac{\sigma : [\![X; \Theta \vdash P[s/x], \Gamma]\!]}{\sigma \circ \pi_s \circ \mathsf{dist}_\Gamma : [\![X; \Theta \vdash \exists x.P, \Gamma]\!]} \quad FV(s) \subseteq X$$

$$\mathsf{P_{\forall}^{\mathsf{T}}} \quad \frac{\sigma : [\![X; \Theta \vdash M, \Gamma, \forall x.N, \Delta]\!]}{[\![\Delta]\!]^-(\mathsf{id} \oslash \pi_s) \circ \sigma : [\![X; \Theta \vdash M, \Gamma, N[s/x], \Delta]\!]} \quad FV(s) \subseteq X$$

$$\mathsf{P_{\forall}^{\mathsf{T}}} \quad \frac{\sigma : [\![X; \Theta \vdash Q, \Gamma, \forall x.N, \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\pi_s \multimap \mathsf{id}) : [\![X; \Theta \vdash Q, \Gamma, N[s/x], \Delta]\!]} \quad FV(s) \subseteq X$$

$$\mathsf{P_{\exists}^{\mathsf{T}}} \quad \frac{\sigma : [\![X; \Theta \vdash M, \Gamma, P[s/x], \Delta]\!]}{[\![\Delta]\!]^-(\pi_s \multimap \mathsf{id}) \circ \sigma : [\![X; \Theta \vdash M, \Gamma, \exists x.P, \Delta]\!]} \quad FV(s) \subseteq X$$

$$\mathsf{P_{\exists}^{\mathsf{T}}} \quad \frac{\sigma : [\![X; \Theta \vdash Q, \Gamma, P[s/x], \Delta]\!]}{\sigma \circ [\![\Delta]\!]^+(\mathsf{id} \oslash \pi_s) : [\![X; \Theta \vdash Q, \Gamma, \exists x.P, \Delta]\!]} \quad FV(s) \subseteq X$$

We show that the core rules suffice to represent any uniform winning strategy σ on a type object provided σ is *bounded* — i.e. there is a bound on the size of plays occurring in σ. In particular, such a strategy is the semantics of a unique *analytic* proof — a proof using only the core rules, with some further restrictions on the use of the matching rule. Given a sequent $X; \Theta \vdash \Gamma$, we say Θ is *lean* if it contains $x \neq y$ for all distinct x and y in X and does not contain $x \neq x$. We assume an arbitrary ordering on variables.

Definition A proof in WS1 is *analytic* if it uses only core rules and has the following additional restrictions:

- Rules other than $\mathsf{P_{\neq}}$ and $\mathsf{P_{ma}^{x,y,z}}$ can only conclude sequents with a lean Θ

- If $\mathsf{P_{ma}^{x,y,z}}$ is used to conclude $X; \Theta \vdash \Gamma$ then Θ does not contain $w \neq w$ for any w; (x, y) is the least pair with $x, y \in X$, $x \not\equiv y$ and $x \neq y \notin \Theta$; and z is the least variable in $\mathsf{Fr}(X; \Theta \vdash \Gamma)$ (the least fresh variable).

Theorem 7.1. *Let $X; \Theta \vdash \Gamma$ be a sequent of WS1 and σ a bounded uniform winning*

strategy on $[\![X;\Theta \vdash \Gamma]\!]$. Then there is a unique analytic proof p of $X;\Theta \vdash \Gamma$ with $[\![p]\!] = \sigma$.

All strategies on the denotations of exponential-free sequents are bounded. Consequently, in the affine fragment we can perform reduction-free normalisation from proofs to (cut-free) core proofs, by reification of their semantics. We thus see that all of the non-core rules are admissible (when restricted to this fragment).

The rest of this section sketches the proof of this full completeness result, and describes an extension to reify unbounded strategies as *infinitary* analytic proofs. We perform a semantics-guided proof search procedure, following [14, 4, 27, 31].

7.1. Uniform Choice

When constructing a proof of a given sequent out of core rules there is a choice of which rule to use when the outermost head connective is $\oplus$ (either $\mathsf{P}_\oplus^1$ or $\mathsf{P}_\oplus^2$) or $\exists$ (which s to use in $\mathsf{P}_\exists^s$). Our choice of rule will depend on the given strategy, depending on which component Player plays in first. However, the input to our procedure is a family of strategies, and we need to ensure that the same component choice is made in each strategy. We will next show that our uniformity condition ensures this.

Proposition 7.2. *If Θ is lean and $(L,v), (M,w) \in \mathcal{M}_X^\Theta$ there exists an $\mathcal{L}$-model $(L,v) \sqcup (M,w)$ with maps $f_{(L,v,M,w)} : (L,v) \to (L,v) \sqcup (M,w)$ and $g_{(L,v,M,w)} : (M,w) \to (L,v) \sqcup (M,w)$.*

Proof If (L,v) is an $\mathcal{L}$-model, define $U_{(L,v)}$ to be the elements of $|L|$ not in the image of v. Then the carrier of $(L,v) \sqcup (M,w)$ is defined to be $X \uplus U_{(L,v)} \uplus U_{(M,w)}$. The $\mathcal{L}$-structure validates all positive atoms, and the valuation is just inj_1. Then the map $f_{(L,v,M,w)}$ sends $v(x)$ to $\mathsf{inj}_1(x)$ and $u \in U_{(L,v)}$ to $\mathsf{inj}_2(u)$. This is an injection because Θ is lean. $g_{(L,v,M,w)}$ is defined similarly. $\square$

We also recall that if $f : (L,v) \to (M,w)$ then $\sigma_{(L,v)}$ is determined entirely by f and $\sigma_{(M,w)}$. In particular, uniformity for positive strategies $\sigma : N \Rightarrow o$ requires that $\sigma_{(L,v)} \sqsubseteq \sigma_{(M,w)} \circ N(f)$ but since $\sigma_{(L,v)}$ is total, it is maximal in the ordering and so we must have $\sigma_{(L,v)} = \sigma_{(M,w)} \circ N(f)$.

Proposition 7.3. *Let $X;\Theta \vdash \Gamma$ be a sequent and suppose Θ is lean. Then there exists an object in $\mathcal{M}_X^\Theta$.*

Proof Note that Θ just contains positive atoms. We can take (X, id), with $(X, \mathsf{id}) \models \overline{\phi}(\overrightarrow{x})$ just if $\overline{\phi}(\overrightarrow{x}) \in \Theta$. Then each formula in Θ is satisfied: each such formula is either $\overline{\phi}(\overrightarrow{x})$, or $x \neq y$ for distinct x, y. $\square$

We now use the above lemmas to show that in any uniform winning strategy on a sequent whose head formula is $P \oplus Q$, either all strategies play their first move in P, or all strategies play their first move in Q.

Proposition 7.4. *Let $M_1, M_2 : \mathcal{M}_X^\Theta \to \mathcal{G}_e$. Suppose Θ is lean, and let $\sigma : M_1 \times M_2 \Rightarrow o$ be a uniform total (resp. winning) strategy. Then $\sigma = \tau \circ \pi_1$ for some uniform total (resp. winning) strategy $\tau : M_1 \Rightarrow o$, or $\sigma = \tau \circ \pi_2$ for some uniform total (resp. winning) strategy $\tau : M_2 \Rightarrow o$.*

Proof We know that each $\sigma_{(L,v)}$ is of the form $\tau_{(L,v)} \circ \pi_i$ for some $i \in \{1,2\}$ since in the game $M_1(L,v) \times M_2(L,v) \multimap o$ we must respond to the initial Opponent-move either with a move in M_1 or a move in M_2 (the π-atomicity condition). But we need to check that i is uniform across components. Suppose that i is not uniform — then we have (L,v) and (T,w) with $\sigma_{(L,v)} = \tau_{(L,v)} \circ \pi_1$ and $\sigma_{(T,w)} = \tau_{(T,w)} \circ \pi_2$. Now consider $(L,v) \sqcup (T,w)$ and let k be such that $\sigma_{(L,v) \sqcup (T,w)} = \tau_{(L,v) \sqcup (T,w)} \circ \pi_k$. By uniformity and totality, $\sigma_{(L,v)} = \sigma_{(L,v) \sqcup (T,w)} \circ (M_1 \times M_2)(f_{(L,v,T,w)}) = \tau_{(L,v) \sqcup (T,w)} \circ \pi_k \circ (M_1 \times M_2)(f_{(L,v,T,w)}) = \tau_{(L,v) \sqcup (T,w)} \circ M_k(f_{(L,v,T,w)}) \circ \pi_k$. But since $\sigma_{(L,v)}$ is of the form $\tau_{(L,v)} \circ \pi_1$, we must have $k = 1$. But we can reason similarly using $\sigma_{(T,w)}$ and $g_{(L,v,T,w)}$ and discover that $k = 2$. This is a contradiction.

Thus there is some i such that each $\sigma_{(L,v)}$ can be decomposed into $\tau_{(L,v)} \circ \pi_i$. In particular, we can take i such that $\sigma_{(X,\mathsf{id})} = \tau_{(X,\mathsf{id})} \circ \pi_i$ where (X, id) is as defined in Proposition 7.3. We only need to show that τ is lax natural. We can construct a natural transformations $\iota_1 : \langle \mathsf{id}, \epsilon \rangle : M_1 \to M_1 \times M_2$ and $\iota_2 : \langle \epsilon, \mathsf{id} \rangle : M_2 \to M_1 \times M_2$. Then $\tau = \sigma \circ \iota_i$, and so is lax natural. $\qquad\square$

We next show that in any uniform family of winning strategies on a sequent with head $\exists x.P$, Player chooses the same x in each strategy component. Moreover, the chosen x is the value of some variable in scope.

Proposition 7.5. *Let $M : \mathcal{M}^\Theta_{X \uplus \{x\}} \to \mathcal{G}_e$. Suppose Θ is lean, and let $\sigma : \forall x.M \Rightarrow o$ be a uniform total (resp. winning) strategy. Then there exists a unique variable $y \in X$ and uniform total (resp. winning) strategy $\tau : M\mathsf{set}^x_y \Rightarrow o$ such that $\sigma = \tau \circ \pi_y$.*

Proof We firstly show that given any $\mathcal{L}$-model (L,v) there is some x with $\sigma_{(L,v)} = \tau_{(L,v)} \circ \pi_{v(x)}$. Suppose for contradiction that $\sigma_{(L,v)} = \tau_{(L,v)} \circ \pi_u$ for some $u \in U_{(L,v)}$. Build the $\mathcal{L}$-model $L' = X \uplus \{a, b\} \uplus U_{(L,v)}$ with valuation inj_1 and validating all positive atoms. Let $\sigma_{(L',\mathsf{inj}_1)} = \tau_{(L',\mathsf{inj}_1)} \circ \pi_r$. Define $m_1 : (L,v) \to (L', \mathsf{inj}_1)$ sending $v(x)$ to $\mathsf{inj}_1(x)$, u to $\mathsf{inj}_2(a)$ and $v \in U_{(L,v)} - \{u\}$ to $\mathsf{inj}_3(v)$. Then $\sigma_{(L,v)} = \sigma_{(L',\mathsf{inj}_1)} \circ \pi_r \circ \forall x.M(m_1)$.

- If $r = \mathsf{inj}_2(b)$ then this is $\sigma_{(L',\mathsf{inj}_1)} \circ \epsilon$ which is ϵ as $\sigma_{(L',\mathsf{inj}_1)}$ must be strict (as its total and a map into o). This is impossible.

- If $r = \mathsf{inj}_1(x)$ then this is $\sigma_{(L',\mathsf{inj}_1)} \circ M(m_1) \circ \pi_{v(x)}$, which is impossible by assumption.

- Hence we must have $r = \mathsf{inj}_2(a)$.

Define $m_2 : (L,v) \to (L', \mathsf{inj}_1)$ sending $v(x)$ to $\mathsf{inj}_1(x)$, u to $\mathsf{inj}_2(b)$ and $v \in U_{(L,v)} - \{u\}$ to $\mathsf{inj}_3(v)$. We can use similar reasoning to show that $r = \mathsf{inj}_2(b)$. This is a contradiction.

Hence, given any (L,v) there is some variable x such that $\sigma_{(L,v)} = \tau_{(L,v)} \circ \pi_{v(x)}$. Let $y \in X$ be the unique variable such that $\sigma_{(X,\mathsf{id})} = \tau_{(X,\mathsf{id})} \circ \pi_y$ where (X, id) is constructed as in Proposition 7.3. We now show the stronger fact that $\sigma_{(L,v)} = \tau_{(L,v)} \circ \pi_{v(y)}$. Suppose that $\sigma_{(L,v)} = \tau_{(L,v)} \circ \pi_{v(x)}$ and $\sigma_{(L,v) \sqcup (X,\mathsf{id})} = \tau_{(L,v) \sqcup (X,\mathsf{id})} \circ \pi_{\mathsf{inj}_1(z)}$. By lax naturality, $\tau_{(L,v)} \circ \pi_{v(x)} = \sigma_{(L,v)} = \sigma_{(L,v) \sqcup (X,\mathsf{id})} \circ \forall x.M(f_{(L,v,X,\mathsf{id})}) = \tau_{(L,v) \sqcup (X,\mathsf{id})} \circ \pi_{v(z)} \circ \forall x.M(f_{(L,v,X,\mathsf{id})})$. Since $\mathsf{inj}_1(z) = f_{(L,v,X,\mathsf{id})}(v(z))$, we have $\sigma_{(L,v)} = \tau_{(L,v) \sqcup (X,\mathsf{id})} \circ M(f_{(L,v,X,\mathsf{id})}) \circ \pi_{v(z)}$ and so we must have $x = z$. By similar reasoning using $g_{(L,v,X,\mathsf{id})}$, we see that $y = z$, so $x = y$.

Hence there is a variable y such that for all (L,v), $\sigma_{(L,v)} = \tau_{(L,v)} \circ \pi_{v(y)}$ for some $\tau_{(L,v)} : M(L, v[x \mapsto v(y)]) \Rightarrow o$. Since Θ is lean, y is the unique variable such that

$\sigma_{(L,v)} = \tau_{(L,v)} \circ \pi_{v(y)}$. Note that $M(L, v[x \mapsto v(y)]) = M(\mathsf{set}_y^x(L, v))$. We can easily check that the resulting transformation $\tau : M\mathsf{set}_y^x \Rightarrow o$ is lax natural. $\qquad \square$

7.2. Reification of Strategies

We define a procedure reify which transforms a bounded uniform winning strategy on a formula object into a proof of that formula. It may be seen as a semantics-guided proof search procedure: given such a strategy σ on the interpretation of Γ, reify finds a proof which denotes it. Reading upwards, the procedure first decomposes the head formula into a unit (nullary connective) using the head introduction rules. If this unit is $\mathbf{1}$, we are done. It cannot be $\mathbf{0}$, as there are no (total) strategies on this game. If the unit is $\top$ or $\bot$, the procedure then consolidates the tail of Γ into a single formula, using the core elimination rules. Once this is done, the head unit is removed using $\mathsf{P}_\bot^+$ or $\mathsf{P}_\top^-$, strictly decreasing the size of the sequent. These steps are then repeated until termination. We further have to deal with equality: whenever a free variable is introduced, we must consider if it is equal to each of the other free variables using the P_{ma} rule.

Informally, if Θ is not lean:

- If Θ contains $x \neq x$ we use $\mathsf{P}_{\neq}$ and halt.

- Otherwise, we consider the least two variables $x, y \in X$ that are not declared distinct by Θ and split the family into those models that identify x and y, and those that do not. In the former case, we can substitute fresh z for both x and y. We then apply the inductive hypothesis to both halves and apply $\mathsf{P}_{\mathsf{ma}}^{x,y,z}$ using $H_{x,y,z}^{-1}$.

If Θ is lean, then:

- The case $\Gamma = \mathbf{0}, \Gamma'$ is impossible: there are no total strategies on this game.

- If $\Gamma = \mathbf{1}, \Gamma'$ then σ must be the empty strategy, since it is the unique total strategy on this game. This is the interpretation of the proof $\mathsf{P_1}$.

- If $\Gamma = \top$ then σ must similarly be the unique total strategy on this game, i.e. the interpretation of $\mathsf{P}_\top$.

- If $\Gamma = \top, P, \Gamma'$ then σ can never play in P since if it did the play restricted to $\top, P$ would not be alternating. Thus σ is a strategy on $\top, \Gamma'$. We can call reify inductively yielding a proof of $\vdash \top, \Gamma'$, and apply $\mathsf{P}_\top^+$ to yield a proof of $\top, P, \Gamma$.

- If $\Gamma = \top, N, P, \Gamma'$ then σ is a total strategy on $\top, N \lhd P, \Gamma$ up to retagging and we can proceed inductively using $\mathsf{P}_\top^{\lhd}$. If $\Gamma = \top, N, M, \Gamma'$ we can proceed similarly, using $\mathsf{P}_\top^{\otimes}$.

- If $\Gamma = \top, N$ then σ is a total strategy on $\downarrow N$: we can strip off the first move yielding a total strategy on N, apply reify inductively yielding a proof of $\vdash N$, and finally apply $\mathsf{P}_\top^-$ yielding a proof of $\vdash \top, N$.

- The case $\Gamma = \bot$ is impossible: there are no total strategies on this game. Other cases where $\bot$ is the head formula proceed as with $\top$: if the tail is a single positive formula, we remove the first move and apply $\mathsf{P}_\bot^+$, otherwise we shorten the tail using $\mathsf{P}_\bot^-$, $\mathsf{P}_\bot^{\oslash}$ or $\mathsf{P}_\bot^{\otimes}$.

- If $\Gamma = A \oslash N, \Gamma'$ then σ is also a strategy on A, N, Γ. We can call reify inductively yielding a proof of $\vdash A, N, \Gamma$ that denotes σ, and apply $\mathsf{P}_\oslash$. We can proceed similarly in the following case $\Gamma = A \lhd P, \Gamma'$.

- If $\Gamma = M \& N, \Gamma'$ then we can split σ into those plays that start with M and those that start with N. This yields total strategies on M, Γ and N, Γ respectively, which we can reify inductively and apply $\mathsf{P}_\&$.

- If $\Gamma = M \otimes N, \Gamma'$ then we can split σ into those plays that start with M and those that start with N. This yields total strategies on M, N, Γ and N, M, Γ respectively, which we can reify inductively and apply $\mathsf{P}_\otimes$.

- If $\Gamma = P \oplus Q, \Gamma$ then σ specifies a first move that must either be in P or in Q. In the former case, we have a strategy on P, Γ and can reify inductively, finally applying $\mathsf{P}_{\oplus_1}$. In the latter case, we have a strategy on Q, Γ and can reify inductively and apply $\mathsf{P}_{\oplus_2}$. The case of $\Gamma = P \,⅋\, Q, \Gamma$ is similar.

- If the head formula is a positive atom $\overline{\phi}(\overrightarrow{x})$ then we must have $\overline{\phi}(\overrightarrow{x})$ in Θ, as otherwise there can be no uniform winning strategies on $[\![\Gamma]\!]$ (since some games in that family have no winning strategies). Thus we can proceed inductively and apply $\mathsf{P}_{\mathsf{at}+}$.

- If the head formula is a negative atom $\phi(\overrightarrow{x})$ then we can split the family σ into those models that satisfy $\phi(\overrightarrow{x})$ and those that do not. All strategies in the latter group must be empty, as there are no moves to play. All strategies in the former group form a uniform winning strategy on $[\![\Theta, \phi(\overrightarrow{x}) \vdash \perp, \Gamma]\!]$ and we can proceed inductively using $\mathsf{P}_{\mathsf{at}-}$.

- If $\sigma : [\![X; \Theta \vdash \Gamma = \forall x.N, \Gamma']\!]$ then $\mathsf{dist}_{\Gamma'} \circ \sigma : I \Rightarrow \forall x.[\![N, \Gamma']\!]$. Using our adjunction, this corresponds to a map $\eta \circ U'_x(\mathsf{dist}_{\Gamma'} \circ \sigma) : I \Rightarrow [\![N, \Gamma']\!]$ in $\mathcal{W}^{\mathcal{M}^\Theta_{X \uplus \{x\}}}$. We can then reify this inductively to yield a proof of $X \uplus \{x\}; \Theta \vdash N, \Gamma'$ and apply $\mathsf{P}_\forall$.

- If $\Gamma = \exists x.P, \Gamma'$ then $\sigma \circ \mathsf{dist}_{+,\Gamma'} : \forall x.[\![P, \Gamma']\!] \Rightarrow o$. By Proposition 7.5, there is a unique y and natural transformation $\tau : [\![P, \Gamma']\!]\mathsf{set}^x_y \Rightarrow o$ such that $\sigma \circ \mathsf{dist}_{+,\Gamma'} = \tau \circ \pi_y$. Since x does not occur in Γ, we have $[\![P, \Gamma']\!]\mathsf{set}^x_y = [\![P[y/x], \Gamma']\!]$. This yields a lax natural transformation $[\![P[y/x], \Gamma']\!] \Rightarrow o$. We can then apply the inductive hypothesis use the $\mathsf{P}^y_\exists$ rule.

We will later show that reify is well founded by giving a measure on sequents that decreases on each call to the inductive hypothesis.

7.3. Definition of Reify

reify$_\Gamma$ is defined inductively in Figure 14. Following the above remarks, the following properties hold:

1a The unique map $i : \varnothing \Rightarrow \mathcal{C}(I, o)$ is a bijection.

1b The map $\mathsf{d} = [\lambda f.f \circ \pi_1, \lambda g.g \circ \pi_2] : \mathcal{C}(M, o) + \mathcal{C}(N, o) \Rightarrow \mathcal{C}(M \times N, o)$ is a bijection. (π-*atomicity* [1]).

2 The map $\_ \multimap o : \mathcal{C}(I, M) \Rightarrow \mathcal{C}(M \multimap o, I \multimap o)$ is a bijection.

For non-lean Θ:

$$\mathsf{reify}_{X,x\neq x;\Theta\vdash\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\neq}$$

$$\mathsf{reify}_{X,x,y;\Theta\vdash\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\mathsf{ma}}^{x,y,z}(\mathsf{reify}(\sigma|_{\mathcal{M}_X^{\Theta,x=y}} \circ H_{x,y,z}^{-1}), \mathsf{reify}(\sigma|_{\mathcal{M}_X^{\Theta,x\neq y}}))$$
$$\text{if } (x,y) \in X \times X \text{ is least such that } x \not\equiv y \text{ and } (x \neq y) \notin \Theta$$
$$\text{and } z \text{ is the least element in } \mathsf{Fr}(X;\Theta\vdash\Gamma)$$

For lean Θ:

$$\mathsf{reify}_{X;\Theta\vdash\phi(\vec{x}),\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\mathsf{at}-}(\mathsf{reify}(\sigma|_{\mathcal{M}_X^{\Theta,\overline{\phi}(\vec{x})}}))$$

$$\mathsf{reify}_{X;\Theta\vdash\overline{\phi}(\vec{x}),\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\mathsf{at}+}(\mathsf{reify}(\sigma))$$

$$\mathsf{reify}_{X;\Theta\vdash\forall x.N,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\forall}(\mathsf{reify}(\eta \circ U_x'(\mathsf{dist}_{\Gamma'} \circ \sigma)))$$

$$\mathsf{reify}_{X;\Theta\vdash\exists x.N,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\exists}^y(\mathsf{reify}(\tau))\text{where } \sigma \circ \mathsf{dist}_{\Gamma}^{-1} = \tau \circ \pi_y$$

$$\mathsf{reify}_{\mathbf{1},\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\mathbf{1}}$$

$$\mathsf{reify}_{\perp,N,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\perp}^-(\mathsf{reify}_{\perp,\Gamma}(\llbracket\Gamma\rrbracket^-(\mathsf{abs}) \circ \sigma))$$

$$\mathsf{reify}_{\perp,P}(\sigma) \quad = \quad \mathsf{P}_{\perp}^+(\mathsf{reify}_P(\Lambda_I^{-1}(\sigma)))$$

$$\mathsf{reify}_{\perp,P,Q,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\perp}^{\otimes}(\mathsf{reify}_{\perp,P\otimes Q,\Gamma}(\llbracket\Gamma\rrbracket^-((\mathsf{sym}\multimap\mathsf{id})\circ\mathsf{pasc}_{\multimap}^{-1}\circ\sigma)))$$

$$\mathsf{reify}_{\perp,P,N,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\perp}^-(\mathsf{reify}_{\perp,P\oslash N,\Gamma}(\llbracket\Gamma\rrbracket^-(\mathsf{lfe})\circ\sigma))$$

$$\mathsf{reify}_{M\&N,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\&}(\mathsf{reify}_{M,\Gamma}(\pi_1\circ\mathsf{dist}_{-,\Gamma}\circ\sigma),\mathsf{reify}_{N,\Gamma}(\pi_2\circ\mathsf{dist}_{-,\Gamma}\circ\sigma))$$

$$\mathsf{reify}_{M\otimes N,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\otimes}(\mathsf{reify}_{M,N,\Gamma}(\pi_1\circ\sigma'),\mathsf{reify}_{N,M,\Gamma}(\pi_2\circ\sigma'))$$
$$\text{where } \sigma' = \mathsf{dist}_{-,\Gamma}\circ\llbracket\Gamma\rrbracket^-(\mathsf{dec})\circ\sigma$$

$$\mathsf{reify}_{A\oslash N,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\oslash}(\mathsf{reify}_{A,N,\Gamma}(\sigma))$$

$$\mathsf{reify}_{A\lhd P,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\lhd}(\mathsf{reify}_{A,P,\Gamma}(\sigma))$$

$$\mathsf{reify}_{\top}(\sigma) \quad = \quad \mathsf{P}_{\top}$$

$$\mathsf{reify}_{\top,P,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\top}^-(\mathsf{reify}_{\perp,\Gamma}(\sigma\circ\llbracket\Gamma\rrbracket^+(\mathsf{abs}^{-1})))$$

$$\mathsf{reify}_{\top,N}(\sigma) \quad = \quad \mathsf{P}_{\top}^-(\mathsf{reify}_N((\_\multimap o)^{-1}(\mathsf{unit}_{\multimap}^{-1}\circ\sigma)))$$

$$\mathsf{reify}_{\top,N,M,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\top}^{\otimes}(\mathsf{reify}_{\top,N\otimes M,\Gamma}(\sigma\circ\llbracket\Gamma\rrbracket^+(\mathsf{pasc}_{\multimap}\circ(\mathsf{sym}\multimap\mathsf{id}))))$$

$$\mathsf{reify}_{\top,N,P,\Gamma}(\sigma) \quad = \quad \mathsf{P}_{\top}^{\lhd}(\mathsf{reify}_{\top,N\lhd P,\Gamma}(\sigma\circ\llbracket\Gamma\rrbracket^+(\mathsf{lfe}^{-1})))$$

$$\mathsf{reify}_{P\oplus Q,\Gamma}(\sigma) \quad = \quad [\mathsf{P}_{\oplus 1}\circ\mathsf{reify}_{P,\Gamma},\mathsf{P}_{\oplus 2}\circ\mathsf{reify}_{Q,\Gamma}]\circ\mathsf{d}^{-1}(\sigma\circ\mathsf{dist}_{+,\Gamma}^{-1})$$

$$\mathsf{reify}_{P\otimes Q,\Gamma}(\sigma) \quad = \quad [\mathsf{P}_{\otimes 1}\circ\mathsf{reify}_{P,Q,\Gamma},\mathsf{P}_{\otimes 2}\circ\mathsf{reify}_{Q,P,\Gamma}]\circ\mathsf{d}^{-1}(\sigma\circ\llbracket\Gamma\rrbracket^+(\mathsf{dec}^{-1})\circ\mathsf{dist}_{+,\Gamma}^{-1})$$

$$\mathsf{reify}_{!N,\Gamma}(\sigma) \quad = \quad \mathsf{P}_!(\mathsf{reify}_{N,!N,\Gamma}(\llbracket\Gamma\rrbracket^-(\alpha)\circ\sigma))$$

$$\mathsf{reify}_{?P,\Gamma}(\sigma) \quad = \quad \mathsf{P}_?(\mathsf{reify}_{P,?P,\Gamma}(\sigma\circ\llbracket\Gamma\rrbracket^+(\alpha^{-1})))$$

We next argue for termination of our procedure. Intuitively, the full completeness procedure first breaks down the head formula until it is $\bot$ or $\top$. It then uses the core elimination rules to compose the tail into (at most) a single formula. These steps do not increase the size of the strategy. Finally, the head is removed using $\mathsf{P}_\bot^+$ or $\mathsf{P}_\top^-$, strictly reducing the size of the strategy. If Θ is not lean, the number of distinct variable pairs that are not declared distinct in Θ is reduced by using P_{ma}.

Formally, we can see this as a lexicographical ordering of four measures on $\sigma, X, \Theta, \Gamma$:

- The most dominant measure is the length of the longest play in σ.

- The second measure is the length of Γ as a list if the head of Γ is $\bot$ or $\top$, and ∞ otherwise.

- The third measure is the size of the head formula of Γ.

- The fourth measure is

$$|\{(x,y) \in X \times X : x \not\equiv y \land x \neq y \notin \Theta\}|$$

If Θ is lean:

- If $\Gamma = \bot, P$ or $\top, N$ then the first measure decreases in the call to the inductive hypothesis.

- Otherwise, if $\Gamma = A, \Gamma'$ with $A \in \bot, \top$ the first measure does not increase and the second measure decreases.

- If $\Gamma = A, \Gamma'$ with $A \notin \{\bot, \top\}$, the first measure does not increase and either the second or third measure decreases.

If Θ is not lean and the P_{ma} rule is applied, in the call to the inductive hypotheses the first three measures stay the same and the fourth measure decreases.

Thus, the inductive hypothesis is used with a smaller value in the compound measure on $\mathbb{N} \times \mathbb{N} \cup \{\infty\} \times \mathbb{N} \times \mathbb{N}$ ordered lexicographically.

7.5. Soundness and Uniqueness

Lemma 7.6. *For all* $\sigma : \llbracket \vdash \Gamma \rrbracket$ *we have* $\llbracket \mathsf{reify}_\Gamma(\sigma) \rrbracket = \sigma$.

Proof We proceed by induction on our reification measure $\langle |\Gamma|, \mathsf{tl}(\Gamma), \mathsf{hd}(\Gamma) \rangle$ using equations that hold in the categorical model. We perform case analysis on Γ. The calculation is routine, we demonstrate only a few cases.

- If Θ is not lean with $(x,y) \in X \times X$ least such that $x \not\equiv y$ and $(x \neq y) \notin \Theta$ and z is the least element in $\mathsf{Fr}(X; \Theta \vdash \Gamma)$, then $\llbracket \mathsf{reify}(\sigma) \rrbracket$
$= \llbracket \mathsf{P}_{\mathsf{ma}}^{x,y,z}(\mathsf{reify}(\sigma|_{\mathcal{M}_X^{\Theta, x=y}} \circ H_{x,y,z}^{-1}, \mathsf{reify}(\sigma|_{\mathcal{M}_X^{\Theta, x \neq y}}))) \rrbracket$
$= [\llbracket \mathsf{reify}(\sigma|_{\mathcal{M}_X^{\Theta, x=y}} \circ H_{x,y,z}^{-1}) \rrbracket H_{x,y,z}, \llbracket \mathsf{reify}(\sigma|_{\mathcal{M}_X^{\Theta, x \neq y}}) \rrbracket]_{x=y, x \neq y}$
$= [\sigma|_{\mathcal{M}_X^{\Theta, x=y}} \circ H_{x,y,z}^{-1} \circ H_{x,y,z}, \sigma|_{\mathcal{M}_X^{\Theta, x \neq y}}]_{x=y, x \neq y}$
$= [\sigma|_{\mathcal{M}_X^{\Theta, x=y}}, \sigma|_{\mathcal{M}_X^{\Theta, x \neq y}}]_{x=y, x \neq y} = \sigma$.

- If Θ is lean and $\Gamma = \phi(\overrightarrow{x}), \Gamma'$ then

$$\llbracket \mathsf{reify}(\sigma) \rrbracket = \llbracket \mathsf{P}_{\mathsf{at}-}(\sigma|_{\mathcal{M}_X^{\Theta,\overline{\phi}(\overrightarrow{x})}}) \rrbracket = [\sigma|_{\mathcal{M}_X^{\Theta,\overline{\phi}(\overrightarrow{x})}}, \epsilon]_{\overline{\phi}(\overrightarrow{x}),\phi(\overrightarrow{x})}^{-} = \sigma$$

    as we must have $\sigma|_{\mathcal{M}_X^{\Theta,\phi(\overrightarrow{x})}} = \epsilon$ since $\llbracket \phi(\overrightarrow{x}), \Gamma \rrbracket_A$ is the terminal object for each A in $\mathcal{M}_X^{\Theta,\phi(\overrightarrow{x})}$.

- If $\Gamma = \forall x.N, \Gamma'$ then $\llbracket \mathsf{reify}(\sigma) \rrbracket = \llbracket \mathsf{P}_\forall(\mathsf{reify}(\eta \circ U_x'(\mathsf{dist}_{\Gamma'} \circ \sigma))) \rrbracket =$
  $\mathsf{dist}_{\Gamma'}^{-1} \circ \llbracket \mathsf{reify}(\eta \circ \widehat{U_x'(\mathsf{dist}_{\Gamma'}} \circ \sigma)) \rrbracket = \mathsf{dist}_{\Gamma'}^{-1} \circ (\eta \circ \widehat{U_x'(\mathsf{dist}_{\Gamma'}} \circ \sigma)) = \mathsf{dist}_{\Gamma'}^{-1} \circ \mathsf{dist}_{\Gamma'} \circ \sigma =$
  σ as required.

- If $\Gamma = P_1 \bindnasrepma P_2, \Delta$ then $\llbracket \mathsf{reify}_{P_1 \bindnasrepma P_2, \Delta}(\sigma) \rrbracket = \llbracket [\mathsf{P}_{\bindnasrepma 1} \circ \mathsf{reify}_{P_1,P_2,\Delta}, \mathsf{P}_{\bindnasrepma 2} \circ \mathsf{reify}_{P_2,P_1,\Delta}] \circ$
  $\mathsf{d}^{-1}(\sigma \circ \llbracket \Delta \rrbracket^+(\mathsf{dec}^{-1}) \circ \mathsf{dist}_{+,\Delta}^{-1}) \rrbracket$. Suppose $\mathsf{d}^{-1}(\sigma \circ \llbracket \Delta \rrbracket^+(\mathsf{dec}^{-1}) \circ \mathsf{dist}_{+,\Delta}^{-1}) = \mathsf{in}_i(\tau)$,
  so $\tau \circ \pi_i = \sigma \circ \llbracket \Delta \rrbracket^+(\mathsf{dec}^{-1}) \circ \mathsf{dist}_{+,\Delta}^{-1}$.

  If $i = 1$ then $\llbracket [\mathsf{P}_{\bindnasrepma 1} \circ \mathsf{reify}_{P_1,P_2,\Delta}, \mathsf{P}_{\bindnasrepma 2} \circ \mathsf{reify}_{P_2,P_1,\Delta}] \circ \mathsf{d}^{-1}(\sigma \circ \llbracket \Delta \rrbracket^+(\mathsf{dec}) \circ \mathsf{dist}_{+,\Delta}^{-1}) \rrbracket =$
  $\llbracket \mathsf{P}_{\bindnasrepma 1}(\mathsf{reify}_{P_1,P_2,\Delta}(\tau)) \rrbracket = \llbracket \mathsf{reify}_{P_1,P_2,\Delta}(\tau) \rrbracket \circ \llbracket \Delta \rrbracket^+(\mathsf{wk}) =$
  $\llbracket \mathsf{reify}_{P_1,P_2,\Delta}(\tau) \rrbracket \circ \llbracket \Delta \rrbracket^+(\pi_1 \circ \mathsf{dec}) = \tau \circ \llbracket \Delta \rrbracket^+(\pi_1 \circ \mathsf{dec}) = \tau \circ \pi_1 \circ \mathsf{dist}_{+,\Delta} \circ \llbracket \Delta \rrbracket^+(\mathsf{dec}) =$
  $\sigma \circ \llbracket \Delta \rrbracket^+(\mathsf{dec}^{-1}) \circ \mathsf{dist}_{+,\Delta}^{-1} \circ \mathsf{dist}_{+,\Delta} \circ \llbracket \Delta \rrbracket^+(\mathsf{dec}) = \sigma$.

  The case for $i = 2$ is similar. $\square$

Lemma 7.7. *For any analytic proof p of $\vdash \Gamma$ we have $\mathsf{reify}_\Gamma(\llbracket p \rrbracket) = p$.*

Proof We proceed by induction on p. The calculation is routine, we demonstrate only a few cases.

- If $p = \mathsf{P}_\perp^+(p')$ with $\Gamma = \perp, P$ then $\mathsf{reify}_\Gamma(\llbracket p \rrbracket) = \mathsf{P}_\perp^+(\mathsf{reify}_P(\Lambda_I^{-1}(\llbracket p \rrbracket))) =$
  $\mathsf{P}_\perp^+(\mathsf{reify}_P(\Lambda_I^{-1}\Lambda_I \llbracket p' \rrbracket)) = \mathsf{P}_\perp^+(\mathsf{reify}_P(\llbracket p' \rrbracket)) = \mathsf{P}_\perp^+(p') = p$.

- If $p = \mathsf{P}_\&(p_1, p_2)$ with $\Gamma = M \& N, \Delta$ then $\mathsf{reify}_\Gamma(\llbracket p \rrbracket)$
  $= \mathsf{P}_\&(\mathsf{reify}_{M,\Delta}(\pi_1 \circ \mathsf{dist}_{-,\Delta} \circ \llbracket p \rrbracket), \mathsf{reify}_{N,\Delta}(\pi_2 \circ \mathsf{dist}_{-,\Delta} \circ \llbracket p \rrbracket))$
  $= \mathsf{P}_\&(\mathsf{reify}_{M,\Delta}(\pi_1 \circ \mathsf{dist}_{-,\Delta} \circ \mathsf{dist}_{-,\Delta}^{-1} \circ \langle \llbracket p_1 \rrbracket, \llbracket p_2 \rrbracket \rangle), \mathsf{reify}_{N,\Delta}(\pi_2 \circ \mathsf{dist}_{-,\Delta} \circ \mathsf{dist}_{-,\Delta}^{-1} \circ$
  $\langle \llbracket p_1 \rrbracket, \llbracket p_2 \rrbracket \rangle))$
  $= \mathsf{P}_\&(\mathsf{reify}_{M,\Delta}(\llbracket p_1 \rrbracket), \mathsf{reify}_{N,\Delta}(\llbracket p_2 \rrbracket))$
  $= \mathsf{P}_\&(p_1, p_2) = p$.

- If $p = \mathsf{P}_{\bindnasrepma 1}(p')$ with $\Gamma = P_1 \bindnasrepma P_2, \Delta$ then $\mathsf{reify}_\Gamma(\llbracket p \rrbracket) = \mathsf{reify}_\Gamma(\llbracket p' \rrbracket \circ \llbracket \Delta \rrbracket^+(\mathsf{wk})) =$
  $[\mathsf{P}_{\bindnasrepma 1} \circ \mathsf{reify}_{P_1,P_2,\Delta}, \mathsf{P}_{\bindnasrepma 2} \circ \mathsf{reify}_{P_2,P_1,\Delta}] \circ \mathsf{d}^{-1}(\llbracket p' \rrbracket \circ \llbracket \Delta \rrbracket^+(\mathsf{wk}) \circ \llbracket \Delta \rrbracket^+(\mathsf{dec}^{-1}) \circ \mathsf{dist}_{+,\Delta}^{-1}) =$
  $[\mathsf{P}_{\bindnasrepma 1} \circ \mathsf{reify}_{P_1,P_2,\Delta}, \mathsf{P}_{\bindnasrepma 2} \circ \mathsf{reify}_{P_2,P_1,\Delta}] \circ \mathsf{d}^{-1}(\llbracket p' \rrbracket \circ \llbracket \Delta \rrbracket^+(\pi_1) \circ \mathsf{dist}_{+,\Delta}^{-1}) = [\mathsf{P}_{\bindnasrepma 1} \circ$
  $\mathsf{reify}_{P_1,P_2,\Delta}, \mathsf{P}_{\bindnasrepma 2} \circ \mathsf{reify}_{P_2,P_1,\Delta}] \circ \mathsf{d}^{-1}(\llbracket p' \rrbracket \circ \pi_1) = [\mathsf{P}_{\bindnasrepma 1} \circ \mathsf{reify}_{P_1,P_2,\Delta}, \mathsf{P}_{\bindnasrepma 2} \circ \mathsf{reify}_{P_2,P_1,\Delta}] \circ$
  $\mathsf{d}^{-1}(\mathsf{d}(\mathsf{in}_1(\llbracket p' \rrbracket))) = \mathsf{P}_{\bindnasrepma 1}(\mathsf{reify}_{P_1,P_2,\Delta}(\llbracket p' \rrbracket)) = \mathsf{P}_{\bindnasrepma 1}(p') = p$ as required. $\square$

This completes our proof of Theorem 7.1.

7.6. Infinitary Analytic Proofs

We have seen that any bounded winning strategy is the denotation of a unique analytic proof of WS1. We cannot use this to normalise proofs to their analytic form because proofs do not necessarily denote bounded strategies. We will next show that our reification procedure can be extended to winning strategies that may be unbounded, provided the resulting analytic proofs are allowed to be *infinitary* — that is, proofs using the core rules that may be infinitely deep. More precisely, we will show that *total* strategies on a type object correspond precisely to the infinitary analytic proofs. Thus we can normalise any proof of WS1 to an infinitary normal form, by taking its semantics and then constructing the corresponding infinitary analytic proof. Two proofs of WS1 are semantically equivalent if and only if they have the same normal form as an infinitary analytic proof.

7.6.1. Infinitary Proofs as a Final Coalgebra

Let L be a set. Let $\mathcal{T}_L$ denote the final coalgebra of the functor $X \mapsto L \times X^*$ in **Set**. The inhabitants of T_L are L-labelled trees of potentially infinite depth. We let $\alpha : \mathcal{T}_L \to L \times \mathcal{T}_L^*$ describe the arrow part of this final coalgebra: this maps a tree to its label and sequence of subtrees. Given a natural number n, we define a function $N_n : \mathcal{T}_L \to \mathcal{P}(L \times \mathcal{T}_L^*)$, by induction: $N_0(T) = \emptyset$ and $N_{n+1}(T) = \{\alpha(T)\} \uplus \bigcup \{N_n(T') : T' \in \pi_2(\alpha(T))\}$. We define the set of nodes $N(T)$ to be $\{N_n(T) : n \in \mathbb{N}\}$. Let Prf be the set of (names of) proof rules of WS1 and Seq the set of sequents of WS1.

Definition An *infinitary analytic proof* of WS1 is an infinitary proof using only the core rules of WS1. Formally, this is an element T of $\mathcal{I} = \mathcal{T}_{\mathsf{Prf} \times \mathsf{Seq}}$ such that for each node $((\mathsf{P}_x, X; \Theta \vdash \Gamma), c) \in N(T)$ we have $|c| = \mathsf{ar}(\mathsf{P}_x)$ and if $(\pi_2 \circ \pi_1 \circ \alpha)(c_i) = X_i; \Theta_i \vdash \Gamma_i$ then the following is a valid core rule of WS1:

$$\mathsf{P}_x \; \frac{X_1; \Theta_1 \vdash \Gamma_1 \qquad \ldots \qquad X_{|c|}; \Theta_{|c|} \vdash \Gamma_{|c|}}{X; \Theta \vdash \Gamma}$$

We let $\mathcal{I}_\Gamma$ denote the set of infinitary analytic proofs of $\vdash \Gamma$.

Let $\{A_{X;\Theta \vdash \Gamma} : X; \Theta \vdash \Gamma \in \mathsf{Seq}\}$ be family of sets indexed by sequents. We can construct a family of maps $A_{X;\Theta \vdash \Gamma} \to \mathcal{I}_{X;\Theta \vdash \Gamma}$ by giving, for each $X; \Theta \vdash \Gamma$ and $a \in A_{X;\Theta \vdash \Gamma}$, a proof rule that concludes $X; \Theta \vdash \Gamma$ from $X_1; \Theta_1 \vdash \Gamma_1$, ..., $X_n; \Theta_n \vdash \Gamma_n$ and for each i an element $a_i \in A_{X_i;\Theta_i \vdash \Gamma_i}$.

7.6.2. Infinitary Proofs as a Limit of Paraproofs

We can consider an alternative approach for presenting our infinitary analytic proofs. We consider partial proofs, that may "give up" in the style of [12].

Definition An *analytic paraproof* of WS1 is a proof made up of the core proof rules of WS1, together with a *dæmon* rule that can prove any sequent:

$$\frac{}{\Phi \vdash \Gamma} \; \mathsf{P}_\epsilon$$

Note that each analytic proof is also an analytic paraproof. Let $\mathcal{C}_\Gamma$ represent the set of analytic paraproofs of $\vdash \Gamma$. We can introduce an ordering $\sqsubseteq$ on this set, generated from the least congruence with P_ϵ as a bottom element. We can take the completion of $\mathcal{C}_\Gamma$ with respect to ω-chains generating an algebraic cpo $\mathcal{D}_\Gamma$. The maximal elements in this domain are precisely the infinitary analytic proofs $\mathcal{I}_\Gamma$, and the compact elements are the analytic paraproofs $\mathcal{C}_\Gamma$.

7.6.3. Semantics of Infinitary Analytic Proofs

We next describe semantics of infinitary analytic proofs via the semantics of analytic paraproofs.

We can interpret analytic paraproofs as partial strategies. We interpret paraproofs of $X; \Theta \vdash \Gamma$ in $\mathcal{G}^{\mathcal{M}_X^\Theta}$. For the rules other than P_ϵ, we use the fact that $\mathcal{G}^{\mathcal{M}_X^\Theta}$ is a WS!-category. We interpret P_ϵ as the strategy $\{\epsilon\}$ where ϵ denotes the empty play on any game. We can hence interpret a analytic paraproof of $\vdash \Gamma$ as a strategy on $[\![\vdash \Gamma]\!]$.

The category $\mathcal{G}^{\mathcal{M}_X^\Theta}$ is cpo-enriched, with $\sigma \sqsubseteq \tau$ if for each A, $\sigma_A \subseteq \tau_A$ as a set of plays. The bottom element is the uniform strategy that is $\{\epsilon\}$ at each component. Composition, pairing and currying are continuous maps of hom sets; as are the operations used in the first-order structure.

Proposition 7.8. *If p and q are analytic paraproofs of $\vdash \Gamma$ and $p \sqsubseteq q$ then $[\![p]\!] \sqsubseteq [\![q]\!]$.*

Proof A simple induction on the proof rules for WS1, using the fact that composition, pairing and currying are monotonic operations. Note that $[\![-]\!]$ is also strict, as $[\![\mathsf{P}_\epsilon]\!] = \{\epsilon\}$.
□

Hom sets of $\mathcal{G}^{\mathcal{M}_X^\Theta}$ are algebraic domains: each strategy is the limit of its compact (finite) approximants. Our monotonic map $\mathcal{C}_\Gamma \to [\![X; \Theta \vdash \Gamma]\!]$ thus extends uniquely to a continuous map $\mathcal{D}_\Gamma \to [\![X; \Theta \vdash \Gamma]\!]$. By construction this agrees with the semantics given above for analytic paraproofs in $\mathcal{D}_\Gamma$. Given any infinitary analytic proof p if $p \downarrow$ is the set of analytic paraproofs less than p then $[\![p]\!] = \bigsqcup [\![p \downarrow]\!]$ using the cpo structure in $\mathcal{G}^{\mathcal{M}_X^\Theta}$.

We can show that this really does capture the intended semantics of infinitary analytic proofs.

Proposition 7.9. *The equations for the semantics of analytic proofs given in Figures 9, 12 and 13 hold for infinitary analytic proofs.*

Proof We use the fact that the constructs used in the semantics of the core proof rules are continuous. We proceed by case analysis on the proof rule.

We just give an example. In the case of $\mathsf{P}_\otimes$, note that $[\![\mathsf{P}_\otimes(p, q)]\!] = \bigsqcup \{[\![r]\!] : r \sqsubseteq \mathsf{P}_\otimes(p, q)\} = \bigsqcup \{[\![\mathsf{P}_\otimes(p', q')]\!] : p' \sqsubseteq p \wedge q' \sqsubseteq q\} = \bigsqcup \{[\![\Gamma]\!]^-(\mathsf{dec}^{-1}) \circ \mathsf{dist}_{-,\Gamma}^{-1} \circ \langle [\![p']\!], [\![q']\!] \rangle : p' \sqsubseteq p \wedge q' \sqsubseteq q\} = [\![\Gamma]\!]^-(\mathsf{dec}^{-1}) \circ \mathsf{dist}_{-,\Gamma}^{-1} \circ \langle [\![\bigsqcup \{p' : p' \sqsubseteq p\}]\!], [\![\bigsqcup \{q' : q' \sqsubseteq q\}]\!] \rangle = [\![\Gamma]\!]^-(\mathsf{dec}^{-1}) \circ \mathsf{dist}_{-,\Gamma}^{-1} \circ \langle [\![p]\!], [\![q]\!] \rangle$ as required. All other cases are similar.
□

7.6.4. Totality

We need to show that given $p \in \mathcal{I}_\Gamma$, $[\![p]\!]$ is a total uniform strategy. Note that this is not true of arbitrary paraproofs in $\mathcal{D}_\Gamma$, nor is it true for infinite derivations in full WS1 (for example, one could repeatedly apply the $\mathsf{P}_{\mathsf{sym}}$ rules forever).

To show this fact, we first introduce some auxiliary notions.

Definition Let $\sigma : N$ be a strategy on a negative game. We say that σ is n-total if whenever $s \in \sigma \wedge |s| \leq n \wedge so \in P_N \Rightarrow \exists p.sop \in \sigma$. A uniform strategy is n-total if it is pointwise n-total.

It is clear that a strategy is total if and only if it is n-total for each n.

Proposition 7.10. *The following hold:*

1. *If σ is n-total and τ is an isomorphism then $\tau \circ \sigma$ is n-total. If σ is n-total and τ is an isomorphism then $\sigma \circ \tau$ is n-total.*

2. *If $\sigma : A \to B$ and $\tau : A \to C$ are n-total then $\langle \sigma, \tau \rangle$ is also n-total. If $\sigma : A_i \multimap B$ is n-total then $\sigma \circ \pi_i : A_1 \times A_2 \multimap B$ is n-total.*

3. *If $\sigma : A \otimes B \multimap C$ is n-total then $\Lambda(\sigma)$ is n-total. If $\sigma : A \multimap B$ is n-total then $\sigma \multimap \mathsf{id} : (B \multimap o) \multimap (A \multimap o)$ is $(n+2)$-total.*

4. *If σ and τ are n-total, then so is $[\sigma, \tau]_{\mathcal{C}, \mathcal{D}}$, $\sigma \circ H$. If σ is n-total, then so is $\hat{\sigma}$.*

Proof Simple verification. $\square$

Proposition 7.11. *Given any infinitary analytic proof p of $X; \Theta \vdash \Gamma$, $[\![p]\!]$ is total.*

Proof We show that $[\![p]\!]$ is n-total for each n. We proceed by induction on a compound measure.

- Define $\mathsf{tl}^+(A, \Gamma)$ to be the length of Γ as a list if $A = \top$ or ∞ otherwise.

- Define $\mathsf{hd}^+(A, \Gamma)$ to be $|A|$ if A is positive or ∞ otherwise.

- Define $\mathsf{tl}^-(A, \Gamma)$ to be the length of Γ as a list if $A = \bot$ or ∞ otherwise.

- Define $\mathsf{hd}^-(A, \Gamma)$ to be $|A|$ if A is negative or ∞ otherwise.

We proceed by induction on

$$f(n, X, \Theta, \Gamma) = \langle n, \mathsf{tl}^+(\Gamma), \mathsf{hd}^+(\Gamma), \mathsf{tl}^-(\Gamma), \mathsf{hd}^-(\Gamma), \mathsf{L}(X, \Theta) \rangle.$$

We proceed by case analysis on p. If $p = \mathsf{P}_\otimes(p_1, p_2)$ then $[\![\mathsf{P}_\otimes(p_1, q_2)]\!] = [\![\Gamma]\!]^-(\mathsf{dec}^{-1}) \circ \mathsf{dist}^{-1}_{-,\Gamma} \circ \langle [\![p_1]\!], [\![p_2]\!] \rangle$. By by Proposition 7.10 $[\![\mathsf{P}_\otimes(p, q)]\!] = [\![p]\!]$ is n-total. The remaining cases work in an entirely analogous way. For $\mathsf{P}_\bot^+$ we must use the fact that currying is continuous and preserves n-totality. For termination:

- If Θ is not lean, in the call to the inductive hypothesis the first five measures do not increase, and the fifth measure L decreases.

- In the case of $\mathsf{P}_\otimes$, $\mathsf{P}_{\&}$, $\mathsf{P}_!$ the first three measures $(n, \mathsf{tl}^+(\Gamma), \mathsf{hd}^+(\Gamma))$ stay the same and either the fourth measure $\mathsf{tl}^-(\Gamma)$ decreases, or the fourth measure stays the same and the fifth measure $\mathsf{hd}^-(\Gamma)$ decreases.

- In the case of $\mathsf{P}_\bot^\otimes$, $\mathsf{P}_\bot^\varnothing$, $\mathsf{P}_\bot^-$ the first three measures stay the same and the fourth measure decreases.

- In the cases of $P^+_\perp$, $P_\parr$, $P_\oplus$, $P_?$ the first measure n stays the same and either the second measure $\mathsf{tl}^+(\Gamma)$ decreases, or the second measure stays the same and the third measure $\mathsf{hd}^+(\Gamma)$ decreases.

- In the case of $P^\otimes_\top$, $P^\lhd_\top$, $P^+_\top$ the first measure stays the same and the second measure decreases.

- In the case of $P^-_\top$, the first measure decreases. In particular, $[\![P^+_\top(q)]\!] = \mathsf{unit}_{\multimap} \circ ([\![q]\!] \multimap \mathsf{id})$. By induction $[\![q]\!]$ is $(n-2)$-total, and so $[\![q]\!] \multimap \mathsf{id}$ is n-total, and so $[\![p]\!]$ is n-total by Proposition 7.10. $\qquad\square$

Note that there are infinitary analytic proofs that denote strategies that are total, but not winning. For example, there is an infinitary analytic proof of $\vdash \perp, ?(\top \lhd \perp)$ given by $P^+_\perp(h)$ where h is the infinitary analytic proof of $\vdash ?(\top \lhd \perp)$ given by $h = P_?(P_\lhd(P^\lhd_\top(P^-_\top(P_\lhd(P^+_\perp(h))))))$. But there are no winning strategies on this game.

7.6.5. Reification of Total Strategies as Infinitary Analytic Proofs

We next show that any total strategy σ on the denotation of a sequent is the interpretation of a unique infinitary analytic proof $\mathsf{reify}(\sigma)$.

We first define reify for winning strategies. We have seen that we can construct a family of maps $A_{X;\Theta\vdash\Gamma} \to \mathcal{I}_{X;\Theta\vdash\Gamma}$ by giving, for each $X;\Theta \vdash \Gamma$ and $a \in A_{X;\Theta\vdash\Gamma}$, a proof rule that concludes $X;\Theta \vdash \Gamma$ from $X_1;\Theta_1 \vdash \Gamma_1, \ldots, X_n,\Theta_n \vdash \Gamma_n$ and for each i an element $a_i \in A_{X_i;\Theta_i\vdash\Gamma_i}$.

$$
\begin{array}{ccc}
\displaystyle\sum_{X;\Theta\vdash\Gamma\in\mathsf{Seq}} A_\Gamma & \xrightarrow{\ f\ } & (\mathsf{Prf} \times \mathsf{Seq}) \times \big(\displaystyle\sum_{X;\Theta\vdash\Gamma\in\mathsf{Seq}} A_{X;\Theta\vdash\Gamma} \big)^* \\[2.5em]
\big(\!\big\langle f \big\rangle\!\big) \Big\downarrow & & \Big\downarrow \mathsf{id} \times \big(\!\big\langle f \big\rangle\!\big)^* \\[2.5em]
\mathcal{I} & \xrightarrow[\ \alpha\ ]{} & (\mathsf{Prf} \times \mathsf{Seq}) \times \mathcal{I}^*
\end{array}
$$

Note that our reification function reify defined in Figure 14 is exactly of this shape. In this case $A_{X;\Theta\vdash\Gamma}$ is the set of uniform winning strategies on $[\![X;\Theta \vdash \Gamma]\!]$. The function specifies, for each strategy, the root-level proof rule and the derived strategies that are given as input to reify coinductively. In the case that σ is bounded, we have seen that the process terminates and $\mathsf{reify}(\sigma)$ is a finite proof.

In fact, we note that this family of maps are still well defined if $A_{X;\Theta\vdash\Gamma}$ is the set $\mathsf{Tot}_{X;\Theta\vdash\Gamma}$ of uniform *total* strategies on $[\![X;\Theta \vdash \Gamma]\!]$. In particular, the composition of a total strategy and an isomorphism is a total strategy; the composition of a total strategy and a projection is a total strategy; and the completeness axioms in Section 7.3 hold with respect to total strategies. This procedure provides, for each total strategy on $X;\Theta \vdash \Gamma$, a proof rule P_x concluding $X;\Theta \vdash \Gamma$ from $X_1;\Theta_1 \vdash \Gamma_1,\ldots,X_n;\Theta_n \vdash \Gamma_n$ and

total strategies on each $\llbracket X_i; \Theta_i \vdash \Gamma_i \rrbracket$. We write this map as $\mathsf{reif}_{X;\Theta\vdash\Gamma}$.

$$
\begin{array}{ccc}
\displaystyle\sum_{X;\Theta\vdash\Gamma\in\mathsf{Seq}} \mathsf{Tot}_{X;\Theta\vdash\Gamma} & \xrightarrow{\ \mathsf{reif}\ } & (\mathsf{Prf}\times\mathsf{Seq})\times(\displaystyle\sum_{X;\Theta\vdash\Gamma\in\mathsf{Seq}}\mathsf{Tot}_{X;\Theta\vdash\Gamma})^* \\
\mathsf{reify}=(\!|\ \mathsf{reif}\ |\!)\ \Big\downarrow & & \Big\downarrow\ \mathsf{id}\times\mathsf{reify}^* \\
\mathcal{I} & \xrightarrow[\ \ \alpha\ \ ]{} & (\mathsf{Prf}\times\mathsf{Seq})\times\mathcal{I}^*
\end{array}
$$

Thus we can take the anamorphism of this map yielding a map from total strategies on $\llbracket X;\Theta\vdash\Gamma\rrbracket$ to $\mathcal{I}_{X;\Theta\vdash\Gamma}$, as required.

7.6.6. Soundness and Uniqueness

We can show that given any winning strategy σ, $\mathsf{reify}(\sigma)$ is the unique infinitary analytic proof p such that $\llbracket\mathsf{reify}(p)\rrbracket=\sigma$.

For soundness, we first introduce some auxiliary notions.

Definition Let σ and τ be strategies on A. We say that $\sigma =_n \tau$ if each play in σ of length at most n is in τ, and each play in τ of length at most n is in σ.

It is clear that $=_n$ is an equivalence relation, and $\sigma = \tau$ if and only if $\sigma =_n \tau$ for each $n \in \mathbb{N}$. We can lift the relation $=_n$ to uniform total strategies pointwise.

Proposition 7.12. *1. If $\sigma =_n \tau$ and ρ is an isomorphism then $\sigma \circ \rho =_n \tau \circ \rho$. If $\sigma =_n \tau$ and ρ is an isomorphism then $\rho \circ \sigma =_n \rho \circ \tau$.*

2. If $\sigma =_n \tau$ and $\rho =_n \delta$ then $\langle\sigma,\rho\rangle =_n \langle\tau,\delta\rangle$. If $\sigma =_n \tau$ then $\sigma \circ \pi_i =_n \tau \circ \pi_i$.

3. If $\sigma =_n \tau$ then $\Lambda(\sigma) =_n \Lambda(\tau)$. If $\sigma =_n \tau$ then $\sigma \multimap \mathsf{id} =_{n+2} \tau \multimap \mathsf{id}$.

4. If $\sigma_1 =_n \sigma_2$ and $\tau_1 =_n \tau_2$ then $[\sigma_1,\tau_1]_{\mathcal{C},\mathcal{D}} = [\sigma_2,\tau_2]_{\mathcal{C},\mathcal{D}}$. If $\sigma_1 =_n \sigma_2$ then $\sigma_1 \circ H =_n \sigma_2 \circ H$. If $\sigma_1 =_n \sigma_2$ then $\hat{\sigma_1} =_n \hat{\sigma_2}$.

Proof Simple verification. $\qquad\square$

Proposition 7.13. *For every uniform total strategy $\sigma : \llbracket \vdash \Gamma \rrbracket$, $\llbracket\mathsf{reify}(\sigma)\rrbracket = \sigma$.*

Proof We show that for each n, $\llbracket\mathsf{reify}(\sigma)\rrbracket =_n \sigma$. The structure of the induction follows that of Proposition 7.11, lexicographically on

$$\langle n, \mathsf{tl}^+(\Gamma), \mathsf{hd}^+(\Gamma), \mathsf{tl}^-(\Gamma), \mathsf{hd}^-(\Gamma), \mathsf{L}(X,\Theta)\rangle.$$

In each particular case, the reasoning follows the proof of Proposition 7.6 using $=_n$ in the inductive hypothesis rather than $=$, and propagating this to the main equation using Proposition 7.12. In the case of $\Gamma = \top, N$ we use the inductive hypothesis with a smaller n, using the final clause in Proposition 7.12. $\qquad\square$

Proposition 7.14. *Given any infinitary analytic proof p, $\mathsf{reify}(\llbracket p\rrbracket) = p$.*

Proof Since $\mathsf{id} = (\!|\, \alpha \,|\!)$, we know that id is the unique morphism f such that:

$$
\begin{array}{ccc}
\mathcal{I}_\Gamma & \xrightarrow{\;\alpha\;} & (\mathsf{Prf} \times \mathsf{Seq}) \times \mathcal{I}^* \\
f \downarrow & & \downarrow \mathsf{id} \times f^* \\
\mathcal{I}_\Gamma & \xrightarrow{\;\alpha\;} & (\mathsf{Prf} \times \mathsf{Seq}) \times \mathcal{I}^*
\end{array}
$$

Thus to show that $\mathsf{reify} \circ [\![-]\!] = \mathsf{id}$ it is sufficient to show that $\alpha \circ \mathsf{reify} \circ [\![-]\!] = \mathsf{id} \times (\mathsf{reify} \circ [\![-]\!])^* \circ \alpha$, i.e. that for each infinitary analytic proof p we have $\alpha(\mathsf{reify}([\![p]\!])) = (\mathsf{id} \times (\mathsf{reify} \circ [\![-]\!])^*)(\alpha(p))$.

- For binary rules P_x we must show that

$$
\mathsf{reify}([\![\mathsf{P}_x(p_1, p_2)]\!]) = \mathsf{P}_x(\mathsf{reify}([\![p_1]\!]), \mathsf{reify}([\![p_2]\!])).
$$

- For unary rules P_x we must show that $\mathsf{reify}([\![\mathsf{P}_x(p)]\!]) = \mathsf{P}_x(\mathsf{reify}([\![p]\!]))$.

- For nullary rules P_x we must show that $\mathsf{reify}([\![\mathsf{P}_x]\!]) = \mathsf{P}_x$.

For each proof rule, we have already shown this in the proof of Proposition 7.7. Proposition 7.9 ensures that the proof applies in this setting. □

7.6.7. Full Completeness and Normalisation
We have thus shown:

Theorem 7.15. *Each total strategy σ on $\vdash \Gamma$ is the denotation of a unique infinitary analytic proof $\mathsf{reify}(\sigma)$.*

We hence have a bijection between infinitary analytic proofs of a formula, and total strategies on the denotation of that formula, via the semantics. Since any proof in WS1 can be given semantics as a winning strategy, and winning strategies are total, we may reify the semantics of a WS1 proof to generate its infinitary normal form $\mathsf{reify}([\![p]\!])$.

Theorem 7.16. *For each WS1 proof p, there is a unique infinitary analytic proof q such that $[\![p]\!] = [\![q]\!]$.*

Proof Let $q = \mathsf{reify}([\![p]\!])$. Then $[\![q]\!] = [\![\mathsf{reify}([\![p]\!])]\!] = [\![p]\!]$ by Proposition 7.13. If q' is an infinitary analytic proof with $[\![q']\!] = [\![p]\!]$ then $[\![q']\!] = [\![q]\!]$ and so $\mathsf{reify}([\![q']\!]) = \mathsf{reify}([\![q]\!])$ and Proposition 7.14 ensures that $q' = q$. □

While infinitary analytic proofs may denote strategies that are not winning, any infinitary analytic proof generated as a result of the above normalisation denotes a winning strategy. The above result also ensures that proofs p_1 and p_2 in WS1 denote the same strategy if and only if their normal forms (as infinitary analytic proofs) are identical.

8. Further Directions

In this paper, we have given some simple examples of "stateful proofs". We aim to investigate further examples in more expressive logics, and to specify additional properties of programs in more powerful programming languages (such as the games-based language in e.g. [30]). Further extensions to our work which may be required in order to do so include:

- WS1 has been presented as a general first-order logic. By adding axioms, we may specify and study programs in particular domains. For example, can we derive a version of Peano Arithmetic in which proofs have constructive, stateful content (cf [10])?

- Extension with *propositional variables* (and potentially, second-order quantification) would allow generic "copycat strategies" to be captured. On the programming side, this would allow us to model languages with polymorphism.

- We have interpreted the exponentials as greatest fixpoints. Adding general inductive and coinductive types, as in μLJ [8] would extend WS1 to a rich collection of datatypes (including finite and infinite lists, for example).

References

[1] Abramsky, S., 2000. Axioms for definability and full completeness, in: Plotkin, G., Stirling, C.P., Tofte, M. (Eds.), Proof, language, and interaction: essays in honour of Robin Milner. MIT Press, Cambridge, MA, USA, pp. 55–75.

[2] Abramsky, S., Honda, K., McCusker, G., 1998. A fully abstract game semantics for general references, in: LICS '98: Proceedings of the 13th Annual IEEE Symposium on Logic in Computer Science, IEEE Computer Society, Washington, DC, USA. p. 334.

[3] Abramsky, S., Jagadeesan, R., 1994. Games and full completeness for multiplicative linear logic. J. Symb. Logic 59, 543–574.

[4] Abramsky, S., Jagadeesan, R., Malacaria, P., 1995. Full Abstraction for PCF. Information and Computation 163, 409–470.

[5] Abramsky, S., McCusker, G., 1996. Linearity, Sharing and State: a fully abstract game semantics for Idealized Algol with active expressions: Extended Abstract. Electronic Notes in Theoretical Computer Science 3, 2 – 14. Linear Logic 96 Tokyo Meeting.

[6] Andreoli, J.M., 2001. Focussing and proof construction. Annals of Pure and Applied Logic 107, 131 – 163.

[7] Blass, A., 1992. A game semantics for linear logic. Annals of Pure and Applied Logic 56, 183 – 220.

[8] Clairambault, P., 2009. Least and greatest fixpoints in game semantics, in: de Alfaro, L. (Ed.), Foundations of Software Science and Computational Structures. Springer Berlin / Heidelberg. volume 5504 of *Lecture Notes in Computer Science*, pp. 16–31. 10.1007/978-3-642-00596-1-3.

[9] Cockett, J.R.B., Cruttwell, G.S.H., Saff, K., 2010. Combinatorial game categories. Submitted to Mathematical Structures in Computer Science, March 2010.

[10] Coquand, T., 1995. A semantics of evidence for classical arithmetic. Journal of Symbolic Logic 60, 325–337.

[11] Curien, P.L., 1993. On the symmetry of sequentiality, in: Proceedings, 9th International Conference on Mathematical Foundations of Computer Science, Springer-Verlag.

[12] Girard, J.Y., 2001. Locus solum: From the rules of logic to the logic of rules. Mathematical Structures in Computer Science 11, 301–506.

[13] Goyet, A., 2013. The lambda lambda-bar calculus: A dual calculus for unconstrained strategies, in: Principles of Programming Languages (POPL).

[14] Hyland, J.M.E., Ong, C.H.L., 2000. On full abstraction for PCF: I, II, and III. Inf. Comput. 163, 285–408.

[15] Hyland, M., 1997. Game semantics, in: Dybjer, P., Pitts, A.M. (Eds.), Semantics and Logics of Computation. Cambridge University Press. Publications of the Newton Institute, pp. 131–184.

[16] Hyland, M., Schalk, A., 2002. Games on graphs and sequentially realizable functionals. extended abstract, in: Logic in Computer Science, 2002. Proceedings. 17th Annual IEEE Symposium on, pp. 257 – 264.

[17] Janelidze, G., Kelly, G.M., 2001. A note on actions of a monoidal category. Theory and Applications of Categories 9, 61–91.

[18] Japaridze, G., 2003. Introduction to Computability Logic. Annals of Pure and Applied Logic 123, 1 – 99.

[19] Joyal, A., . Remarques sur la théorie des jeux à deux personnes. Gazette des Sciences Mathématiques du Quebec 1, 46–52.

[20] Laird, J., 1997. Full abstraction for functional languages with control, in: In Proceedings, Twelfth Annual IEEE Symposium on Logic in Computer Science, IEEE Computer Society Press. pp. 58–67.

[21] Laird, J., 2003. A categorical semantics of higher order store. Electronic Notes in Theoretical Computer Science 69, 209 – 226. CTCS'02, Category Theory and Computer Science.

[22] Laird, J., 2005. Locally Boolean domains. Theoretical Computer Science 342, 132 – 148. Applied Semantics: Selected Topics.

[23] Laird, J., 2006. A calculus of coroutines. Theoretical Computer Science 350, 275 – 291. Automata, Languages and Programming: Logic and Semantics (ICALP-B 2004).

[24] Laird, J., 2010. Functional programs as coroutines : A semantic analysis.

[25] Lamarche, F., 1992. Sequentiality, games and linear logic, in: Proceedings, CLICS workshop, Aarhus University, DAIMI-397–II.

[26] Lamarche, F., 1995. Games semantics for full propositional linear logic, in: Proceedings, Tenth Annual IEEE Symposium on Logic in Computer Science, IEEE Computer Society Press. pp. 464–473.

[27] Laurent, O., 2002. Polarized games, in: Logic in Computer Science, 2002. Proceedings. 17th Annual IEEE Symposium on, pp. 265–274.

[28] Laurent, O., 2005. Classical isomorphisms of types. Mathematical. Structures in Comp. Sci. 15, 969–1004.

[29] Laurent, O., 2010. Game semantics for first-order logic. Logical Methods in Computer Science 6, 3.

[30] Longley, J., 2009. Some programming languages suggested by game models (extended abstract). Electronic Notes in Theoretical Computer Science 249, 117 – 134. Proceedings of the 25th Conference on Mathematical Foundations of Programming Semantics (MFPS 2009).

[31] Melliès, P., 2005. Asynchronous games 4: a fully complete model of propositional linear logic, in: Logic in Computer Science, 2005. LICS 2005. Proceedings. 20th Annual IEEE Symposium on, IEEE. pp. 386–395.

[32] Melliès, P.A., 2009. Categorical semantics of linear logic. Société Mathématique de France. volume 27 of *Panoramas et Synthèses*. chapter 1. pp. 15–215.

[33] Melliès, P.A., 2012. Game semantics in string diagrams, in: Logic in Computer Science (LICS), 2012 27th Annual IEEE Symposium on, pp. 481 –490.

[34] Melliès, P.A., Tabareau, N., 2010. Resource modalities in tensor logic. Annals of Pure and Applied Logic 161, 632 – 653. The Third workshop on Games for Logic and Programming Languages (GaLoP), Galop 2008.

[35] Melliès, P.A., Tabareau, N., Tasson, C., 2009. An explicit formula for the free exponential modality of linear logic, in: ICALP '09: Proceedings of the 36th Internatilonal Colloquium on Automata, Languages and Programming, Springer-Verlag, Berlin, Heidelberg. pp. 247–260.

[36] Reynolds, J.C., 1981. The essence of Algol, in: Proceedings of the 1981 International Symposium on Algorithmic Languages, North-Holland. pp. 345–372.

[37] Schalk, A., 2004. What is a categorical model of linear logic. Technical Report. Department of Computer Science, University of Manchester.

[38] Wolverson, N., 2008. Game Semantics for an object-oriented language. Ph.D. thesis. University of Edinburgh.