Unifying static and dynamic denotational semantics

Pierre Boudes

Institut de mathématiques de Luminy UMR 6206,
campus de Luminy case 907,
13288 Marseille cedex 9, France,
boudes@iml.univ-mrs.fr,
WWW home page: http://boudes.lautre.net

Abstract. This work deals with semantics of programming languages (or equivalently, thanks to Curry-Howard isomorphism, with semantics of proofs). We introduce a framework in which both static semantics (Ehrhard's hypercoherences) and dynamic semantics (Hyland-Ong's games), can be presented. The work is carried in a multiplicative subsystem of Laurent's polarized linear logic with weakening. Like Böhm trees for lambda-calculus, designs, adapted from Girard's ludics, play the role of intermediate objets between the syntax and the (two) semantics. Our framework allows us to define a new coherence semantics and to prove its full completeness.

1 Introduction

Denotational semantics interprets a *program* (a proof or a λ-term) as a structure representing all its possible interactions (via cut elimination or via β-reduction) with others programs. In such structures (from now on called *semantical agents* or agents) interaction is either represented as a *result* or as a computational trace (an history of the reduction). In the first case the semantics is static and in the second the semantics is dynamic. We are mainly concerned with the comparison of these two kinds of semantics and more precisely, with the comparison of hypercoherences ([Ehr93]), a static semantics refining Scott's domains and coherence spaces where agents are cliques in hypergraphs, and game semantics, a dynamic semantics where agents are *strategies*.

This study is motivated by the result of T. Ehrhard [Ehr96] which surprisingly relates games and hypercoherences by stating that the extensional collapse of sequential algorithms (a game model) is the hypercoherence model. On this issue one can see [Mel03,Bou04].

Denotational semantics must enjoy *soundness*: two proofs equivalent by cut elimination (or two β-equivalent λ-terms) are interpreted as the same semantical agent. When the interpretation map is surjective (each semantical agent is the interpretation of a proof), semantics is *fully complete*. When the interpretation map is injective for a *reasonable quotient* of syntax, semantics is injective.

Linear logic ([Gir87]), LL for short, admits a proof-net syntax which gives a more accurate presentation of proofs than sequent calculus for the purpose of studying injectivity of the semantics.

A static semantics of LL, called the *relational model*, underlies hypercoherences and coherence spaces (where agents are cliques in graphs). In the relational model agents are sets of points. In LL without exponentials, the interpretation of a proof in the relational model, in the coherence space model and in the hypercoherence model coincide.

There are two kinds of games semantics: Abramsky-Jagadeesan-Malacaria's games ([AJM94]) and Hyland-Ong's games ([HO00]). We are only interested in HO games. A HO game consists of: a tree (or forest) order, called the *arena* of the game; a polarity convention which maps odd/even onto opponent/proponent; and a notion of legal plays (or legal positions). In a play, moves are nodes of the arena. Each play has a proponent or opponent polarity depending on the parity of its height in the arena. A HO play is a sequence of alternating moves equipped with *pointers* such that each occurrence of a move b which has a father a in the arena points onto one occurrence of a which appears before b in the play. The pointing relation is used to compute a subsequence of the play: the *proponent's view*. HO semantical agents are innocent strategies: sets of plays whose proponent's views form *deterministic* trees (each two sequences first differ on opponent's moves).

Polarized games. Polarized games are a fully complete HO game semantics of polarized linear logic (LLP), introduced by O. Laurent in [Lau02] and for which he recently proved an injectivity result ([Lau03]).

1.1 Multiplicative polarized linear logic with weakening (WMLLP)

In this paper, we only consider logical systems without atoms.

Formulæ of multiplicative exponential polarized linear logic (MELLP) are given by:

$$N := \bot \mid N \parr N \mid ?P \qquad \text{(negative formulæ)}$$
$$P := 1 \mid P \otimes P \mid !N \qquad \text{(positive formulæ)}$$

with the usual De Morgan laws for the orthogonal $(-)^{\perp}$. Rules of MELLP are given in figure 1. We use Γ to range over contexts and $\mathcal{N}, \mathcal{N}'$ to range over contexts made of negative formulæ.

The logicial system we consider in this paper is WMLLP which consists of MELLP without the contraction rule. WMLLP is not exactly a polarized version of multiplicative affine linear logic since, in WMLLP, like in MELLP, there is no weakening on positive formulæ. In WMLLP '!' and '?' are not real exponentials, they are just shifts of polarities allowing weakening on negative formulæ.

$$\frac{}{\vdash 1}\,(one) \qquad \frac{\vdash N, N', \Gamma}{\vdash N \parr N', \Gamma}\,(par) \qquad \frac{\vdash \Gamma}{\vdash N, \Gamma}\,(weak.) \qquad \frac{\vdash \mathcal{N}, P}{\vdash \mathcal{N}, ?P}\,(der.)$$

$$\left(\frac{\vdash N, N, \Gamma}{\vdash N, \Gamma}\,(cont.)\right)$$

$$\frac{\vdash \mathcal{N}, P \qquad \vdash \mathcal{N}', P'}{\vdash \mathcal{N}, \mathcal{N}', P \otimes P'}\,(tens.) \qquad \frac{\mathcal{N}, N}{\vdash \mathcal{N}, !N}\,(prom.) \qquad \frac{\vdash \mathcal{N}, N^{\perp} \qquad \vdash N, \Gamma}{\vdash \mathcal{N}, \Gamma}\,(cut)$$

Fig. 1. MELLP rules

1.2 The full picture

Our present work relies on three remarks. (The first two remarks work fine in MELLP but for sake of simplicity we don't state them in their full generality.)

Remark 1. **The arena of the polarized game interpretation of a WMLLP formula is the *polarized* syntactical tree structure of this formula**. Moreover every finite arena is given by a formula.

Remark 2. In a polarized game, **each play is a *rooted subtree* of the arena of the game, whose order have been totalized**. And **rooted subtrees are exactly the points of the relational model**.

Remark 3. A strategy is traditionally a set of plays but **an innocent strategy can also be presented as a deterministic tree of proponent's views that is**, following Faggian and Hyland (see [FH02]), **a design in the sense of ludics** (an abstract notion of cut free proof introduced by J.-Y. Girard, see [Gir01]).

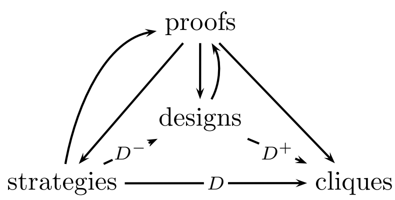

Fig. 2. The unifying framework.

Using these remarks, we introduce a framework based on designs where both static and dynamic semantics are presented. In what follows strategy in HO games will always mean innocent strategy.

Figure 2 gives a picture of how things are related in this framework. The three maps whose source is the set of proofs are interpretation maps. Curved lines express full completeness results. The desequentializing projection D from strategies to cliques factors into two maps: a negative desequentialization D^- from strategies to designs which maps strategies to trees of proponent's views and a positive desequentialization D^+ from designs to cliques. These two maps are injective. The diagram commutes.

1.3 Content of the paper

Section 2 presents basic notions on trees and rooted subtrees.

In section 3, we interpret WMLLP in the relational model, in the coherence space model, in the hypercoherence model and in the polarized game model. We give a new characterization of the coherence relation in the coherence space and hypercoherence interpretations of formulæ which will prove useful in section 5.

In section 4, we show how to associate a design to a cut free proof of WMLLP. We prove a full completeness result and we show how designs relate to polarized game strategies and hypercoherence cliques.

The designs we use here are simpler than the original designs of ludics, mainly because we do not use additive connectives.

In section 5, we introduce a hypercoherence based semantics of WMLLP whose underlying relational model contains only points which come from branches in designs (*chronicles* in the ludics terminology). This semantics enjoys a subdefinability property (each clique is a subset of the interpretation of a proof) and a full completeness result. These last results are important, since there are very few full completeness results for hypercoherence models. There was one by A. Tan for multiplicative linear logic ([Tan97]) and one, generalizing the former, by R. Blute, M. Hamano, P. Scott for multiplicative additive linear logic ([BHS04]). These results require some additional categorical constructions (dinats and *double glueing*).

In what follows polarized means negative or positive. In games negative means opponent and positive means proponent.

2 Trees and subtrees

Multisets. We use the notation [ ] for multisets while the notation { } is, as usual, for sets. The pairwise union of multisets is denoted by a + sign and following this notation the generalized union is denoted by a $\sum$ sign. The neutral element for this operation, the empty multiset, is denoted by []. In this paper, by multiset we always mean finite multiset.

If R is a relation, R^* denotes its transitive closure.

We will use various representations of non-empty finite trees and we will often change from one to the other for a given tree.

Order-theoretic trees. In games semantics litterature, a finite tree t is a partial order $(S_t, \leq_t)$, where S_t is a finite set, having a least element (the *root*) and such that if $a \leq_t c$ and $b \leq_t c$ then $a \leq_t b$ or $b \leq_t a$. The cardinality $\sharp t$ of a tree $t = (S_t, \leq_t)$ is the cardinality of S_t. We sometimes use t to denote the set S_t. Elements of a tree are *nodes* and maximal elements are *leaves*. The associated precedence relation is denoted by $<^1_t$ (so $a <^1_t b$ means that $a <_t b$ and $a \leq_t c \leq_t b \implies c = a$ or $c = b$). The *sons* of a node a are nodes b such that $a <^1_t b$.

A *finitely branching* tree is a tree such that each node has a finite number of sons. An *ordered tree* is a tree together with a total ordering of the sons of a, for each node a. A labeling of nodes in a tree is a function from S_t to a set L of labels. By *labeled trees* we always mean trees with labeling of nodes. A *polarized tree* is a tree together with a polarity, positive or negative. In a polarized tree, each node has a polarity (the polarity of the root is the polarity of the tree itself and polarites alternate from root to leaves). So the polarity defines a set of negative nodes t^- and a set of positive nodes t^+ and also a negative precedence relation $<_t^- = <_t^1 \cap (t^+ \times t^-)$ and a positive precedence relation $<_t^+ = <_t^1 \cap (t^- \times t^+)$.

Graph theory provides equvalent definitions of trees. We just recall that if $a <_t^1 b$ then, in graph theory, one says that there is an *edge* between a and b. There are also representations of trees as sets of words on an alphabet of labels. For instance, if W is a set of words then its *prefix tree* given as a partial order is a labeled tree $(S_t, \leq_t)$ such that S is the set of non-empty prefix of elements of W and $\leq_t$ is the prefix ordering, and where a node $w \cdot l$ is labeled by its last letter l.

Trees as data structures. Sometimes we don't want to explicitly refer to a set of nodes S_t (we work up to isomorphisms). For that purpose, it is convenient to represents finitely branching trees and ordered trees as data stuctures, as follows. A (labeled) finitely branching, ordered tree is a tuple $(t_1, \ldots, t_n)$ of (labeled) finitely branching, ordered trees (and a label). A similar definition holds for unordered trees with multisets instead of tuples. For instance, the empty tuple and the empty multiset represent the equivalence class of trees of one node for both trees and ordered trees isomorphisms relation.

If an ordered tree t is given as a data structure we may want to see it as an order $(S_t, \leq_t)$. For that purpose we fix a canonical choice as follows. The set S_t is a set of words of integers. The root is the empty word and if a node is w then its n sons are, in order, $w \cdot 0, \ldots, w \cdot (n-1)$. The partial order $\leq_t$ is then the prefix order and sons orderings are given by the natural ordering of the integers at the end of words. For a non ordered tree t given by a data structure there is no canonical choice but we can assume orderings of sons, make the canonical choice for $(S_t, \leq_t)$ and forget about orderings of sons.

Which representation of trees we are using depends on the context. The default choice for ordered trees is data structure while the default choice for unordred trees is partial orders.

Definition 4 (Rooted subtrees.). *Let $t = (t_1, \ldots, t_n)$ be a non empty (labeled) ordered tree. A rooted subtree of t is a tuple $(\mu_1, \ldots, \mu_n)$ (and a label l) where for each i, μ_i is either the empty multiset $[]$ or a singleton multiset $[p_i]$ where p_i is a rooted subtree of t_i.*

The use of multisets in this definition is motivated by the study of the general MELLP case where rooted subtrees can be *thick*, that is where multisets involved in subtrees can have more than one element.

We extend this definiton by setting the set of rooted subtrees of the empty tree equal to the empty set. There is no empty rooted subtree. To each rooted subtree p we associate the ordered tree obtained by removing empty multisets and replacing each singleton $[p']$ by p' in each tuple of p.

3 Static and dynamic semantics

We associate to each formula A of WMLLP a polarized, non empty, finite, ordered tree, called the *polarized tree* of the formula, denoted $t(A)$ and defined as follows. The polarity of $t(A)$ is the polarity of A and $t(A^\perp) = t(A)$, with polarities exchanged. The ordered tree $t(1)$ is the empty tuple. The ordered tree $t(A_1 \otimes \ldots \otimes A_n)$ is the tuple equal to the concatenation $t(A_1) \cdot \ldots \cdot t(A_n)$ of the tuples $t(A_i)$. The ordered tree $t(?A)$ is the tuple $(t(A))$.

Polarized trees represent formulæ up to associativity and neutrality of multiplicatives but not up to commutativity. Logical systems we study are commutative. So the real structure we are interested in are unordered trees. But ordered trees are more convenient for dealing with subtrees.

3.1 Relational model

The interpretation of a formula A is a set, denoted $|A|$ and called the *web* of A. The web of a multiplicative constant contains only one element, the empty multiset *which we identify with the empty tuple*. The web of $A_1 \parr \ldots \parr A_n$ (resp. $A_1 \otimes \ldots \otimes A_n$) is the Cartesian product $|A_1| \times \ldots \times |A_n|$. The web of $?A$ (resp. $!A$) is $\{[a] \mid a \in |A|\} \cup \{[]\}$. *Points* in a formula are just the element of the web of this formula.

Proposition 5. *For the WMLLP relational model, the web of a formula A is the set of rooted subtrees of $t(A)$.*

Remark that, for each negative formula N of WMLLP $|N|$ contains a unique element ϵ_N corresponding to the rooted subtree of $t(N)$ equal to its root.

We introduce a convenient semantical operation, the negative shift, denoted $\uparrow$, which we will only apply to the web of positive formulæ ans which is defined by setting $\uparrow |P| = \{[a] \mid a \in |P|\}$.

The interpretation of a proof of $\vdash N_1, \ldots, N_n$ (resp. $\vdash N_1, \ldots, N_n, P$) is a subset of $|N_1| \times \ldots \times |N_n|$ (resp. of $|N_1| \times \ldots \times |N_n| \times |\uparrow P|$).

In the relational model we work up to associativity of the Cartesian product. Interpretation of proofs is given inductively on rules. We follow notations of figure 1, page 3.

The proof consisting of the *one* rule is interpreted as the singleton $\{[[]]\}$. *Par* and dereliction rules leave unchanged the set interpreting the proof.

If the set associated with the premise of a weakening rule is x then the set associated with its conclusion is $x \times \{\epsilon_N\}$.

If x is the set associated with the premise of a promotion rule then we associate to its conclusion the subset $\{(\gamma, [[a]]) \mid (\gamma, a) \in x \text{ and } a \in |N|\} \cup (\epsilon_{\mathcal{N}}, [[]])$ of $|\mathcal{N}| \times \uparrow |!N|$.

If x and y are the sets associated respectively with the first and the second premise of a tensor rule then we associate to its conclusion the set of all tuples $(\gamma, \delta, [(a, a')])$ where $(\gamma, [a]) \in x$ and $(\delta, [a']) \in y$.

If x and y are the sets associated respectively with the first and the second premise of a cut rule then we associate to its conclusion the set of all tuples

(γ, δ) where $(\gamma, [a]) \in x$ and $(a, \delta) \in y$. This is a polarized version of the usual relational composition.

3.2 Coherence spaces and hypercoherences

A coherence space (resp. a hypercoherence) is a reflexive undirected graph (resp. hypergraph). Each formula A is interpreted as a coherence space (resp. hypercoherence) whose set of vertices is the web of A and whose set of edges (resp. hyperedges), called *coherence*. Two (resp. a non empty finite set of) elements of the web are coherent when they form an edge (resp. a hyperedge), and otherwise these points are *strictly incoherent*. Remark that a subset of a hyperedge is not necessarily a hyperedge. Elements of the web which are coherent and not all equal are *strictly coherent*. Coherence is reflexive. Coherence in formulæ is defined as follows. Orthogonal exchanges strict coherence and strict incoherence. A pair (resp. a non finite set) of elements is coherent in a tensor $P \otimes P'$ iff the projection of this pair (resp. this set) on P (which is a pair, resp. a non empty finite set) is coherent and its projection on P' is coherent. If at least one of the elements in a pair (resp. a non empty finite set) of elements of $|!N|$ is the empty multiset then this pair (resp. this set) is coherent, otherwise this pair (resp. this set) is coherent iff $[a] \mapsto a$ maps this pair (resp. this set) on a coherent pair (resp. a coherent set) of A.

Let us recall that, in coherence spaces (resp. hypercoherences), semantical agents of type A are cliques in the graph (resp. hypergraph) interpreting A. A clique is a set x of points in A s.t. each pair (resp. finite set of) elements of x is coherent.

For WMLLP the interpretation of proofs in coherence spaces and hypercoherences is the same as in the relational model.

Remark 6. For WMLLP[1], a pair of points of type A is a clique in the coherence space semantics iff it is a clique in the hypercoherence semantics. This is not the case for MELLP (for instance). Even for WMLLP, the general notions of clique of the two semantics differ: if x is a clique in hypercoherences then x is a clique in coherence spaces but the converse is not true.

Definition 7. *If $p = (\mu_1, \dots, \mu_k)$ and $p' = (\mu'_1, \dots, \mu'_k)$ are two rooted subtrees of a same tree then p is a subtree of p', notation $p \subseteq p'$, iff for each i, if $\mu'_i = []$ then $\mu_i = []$ and if $\mu_i = [q]$ and $\mu'_i = [q']$ then $q \subseteq q'$. For this order, the infimum of a family of rooted subtrees of a same tree is called* intersection *and denoted by an intersection sign, and the supremum is called* superposition *and denoted by a union sign. If $p \subseteq q$ the* frontier *of p in q is the set $F(p, q) = \{b \in p \mid \exists c \in q \setminus p, b <^1_q c\}$ (here, rooted subtrees are seen as partial orders).*

Proposition 8. *If $(p_i)_{i \in I}$ is a non empty finite family of points in a formula A then the set $\{p_i \mid i \in I\}$ is coherent (resp. strictly coherent) iff $\{\cap p_i, \cup p_i\}$ is coherent (resp. strictly coherent). And, if $p \subsetneq q$ are two points of the same type*

[1] Let us recall that we only consider logical systems without atoms.

then $\{p, q\}$ is strictly coherent iff there exists a rooted subtree $t \subseteq p$ such that for each $a \in t$:

$$\exists b \in p, a <_p^1 b \implies \exists c \in t \mid a <_t^1 c; \tag{1}$$

$$a \in t^+ \implies \{b \in p \mid a <_p^1 b\} = \{b \in t \mid a <_t^1 b\}; \tag{2}$$

$$a \in t^- \implies \textit{if } \exists b \in F(p,q) \cap p^-, a \leq_p b \quad \textit{then } \exists c \in F(p,q) \cap t^+, a \leq_t c. \tag{3}$$

Corollary 9. *The coherence of a non empty finite set of points only depends on the intersection and the superposition of these points. If a set of points x is closed under intersection and superposition of trees then x is a clique in hypercoherences iff it is a clique in coherence spaces.*

3.3 Polarized games

According to [Lau03], the arena associated with a formula A is just the polarized tree $t(A)$ of this formula presented as a tree order. We work on arenas up to polarized ordered trees isomorphisms.

We do not recall the original polarized games semantics of LLP ([Lau02]). We just stress how we adapt this semantics to our framework for the reader who already knows polarized games. First, following [Lau03], we do not require *co-visibility* of plays in strategies and second we interpret promotion as the affine promotion denoted $\downarrow$ and called positive shift in [Lau02] (the interpretation of others rules is unchanged).

Definition 10. *A polarized tree t which has as many negative nodes as positive nodes is* balanced*.*

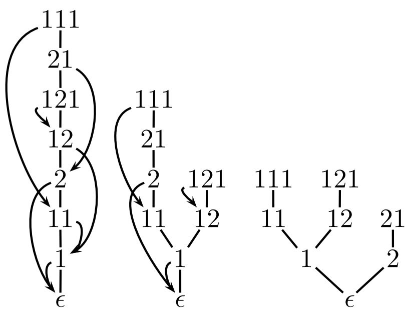

Fig. 3. Example

According to [Lau02], a (legal) play in the interpretation of a proof is *well-opened*: there is only one move which does not point to a preceding move of the play and this is the first move. Moreover with our setting for WMLLP, if s is a play in a strategy interpreting a proof, then s is *repetition-free*: s does not contain two different occurrences of a same move.

We now give an alternative presentation of plays in strategies. A repetition-free well-opened legal play s in a formula A is a rooted subtree p of $t(A)$ whose order have been totalized into a sequence s which alternates between negative and positive nodes of p. This sequence s is equipped with the following pointing relation: b points to a if $a <_p^1 b$. When s is a repetition-free well-opened legal play the underlying subtree p is unique, and we denote it by $d(s)$. If s is of even length then $d(s)$ is balanced. Conversely, if a rooted subtree p of $t(A)$ is balanced then there exists an even length repetition-free well-opened legal play s of A such that $d(s) = p$.

We now define general notions which subsume the standard games notions of opponent's and proponent's view and of visibility and co-visibility.

Definition 11 (Views and negative desequentialization). *Let E be a finite set and let $s = (E, \leq_s)$ and $t = (E, \leq_t)$ be two finite polarized trees of the same polarity and such that $s^- = t^-$, $s^+ = t^+$ and $\leq_t \subseteq \leq_s$. Then the tree $(E, (<_s^+ \cup <_t^-)^*)$ is the* positive view *of s for t, notation $v^+(s,t)$, and the tree $(E, (<_s^- \cup <_t^+)^*)$ is the* negative view *of s for t, notation $v^-(s,t)$. If $<_t^+$ is included in $(<_s^+ \cup <_t^-)^*$ we say that s is* visible *for t and if $<_t^-$ is included in $(<_s^- \cup <_t^+)^*$ we say that s is* co-visible *for t. When speaking of positive (or negative) views if t is not specified then t is assumed to be equal to $d(s)$ and $v^+(s, d(s))$ is denoted by $d^-(s)$ (the positive view is a desequentialization which forget about sequentialization of* negative *nodes).*

Figure 3 shows three trees: the first is a play s, the second consists in $d^-(s)$ decorated by pointers representing $<_{d(s)}^+$ and the third is $d(s)$. For sake of clarity nodes are labeled using the ludics convention of the next section.

4 Designs in WMLLP

Localization of formulæ. Following Ludics, we label nodes of the polarized tree associated to a formula by *loci*. A *locus* is just a word of integers. We label nodes of $t(A)$ by arbitrary loci in such a manner that different nodes have different labels. So we can address nodes of $t(A)$ by their labels. We further require that for each pair of nodes n and m if m is a son of n, the label of m is the concatenation of the label of n and of an integer. Such a labeling of $t(A)$ is called a *localization* of A. And, once given a localization for a formula A, the labeled tree $t(A)$ is called a *localized formula*. A canonical localization is defined following the pattern giving the canonical partial order representing an ordered tree (see section 2). If no localization of A is given the canonical choice is assumed. If required, we can relocalize formulæ.

Addresses of sub-formulæ. Let t be a localization of a formula A and let A' be a sub-formula of A (in the usual sense). Then, to each occurrence of A' in A corresponds a subtree t' of t (we just follow the syntactical strucure of A in $t(A)$). The address of this occurrence of A' in t is just the label of the root of t'.

Definition 12. *A polarized tree is called* deterministic *when each negative node has at most one (positive) son. A tree is called* even *when every path from the root to a leaf is of even length. So an even, negative, deterministic tree is a negative tree such that each negative node has exactly one (positive) son. A rooted subtree t' of a polarized tree t is* total *when all the sons in t of each positive node of t' are in t'.*

Definition 13 (Designs and WMLLP designs). *A* design *is a labeled, even, deterministic, negative, finite tree where labels are such that:*

- *each node but the root is labeled by a locus and the root is labeled by a multiset of loci;*
- *for each positive node n, the labels of the sons of n are pairwise distinct and each one consists of the concatenation of the label of n with an integer;*
- *for each positive node n of t, the label of n is the concatenation of a locus a and of an integer where a is the label of a negative node below n, or a is an element of the multiset labeling the root.*

A WMLLP design *in a localized negative formula t, is a design $\mathcal{D}$ such that:*

- *the multiset labeling the root contains only one locus (that we will identify with the label of the root);*
- *two different nodes of $\mathcal{D}$ have different labels;*
- *and the set of loci occurring in $\mathcal{D}$ is the set of labels of a total rooted subtree t' of t (this tree t' is necessarily unique).*

4.1 Cut free proofs as designs

We now define the design interpretation of cut free proofs. We restrict ourselves to the case of proofs whose conclusion sequent contains only one negative formula. The general cut free proofs case, which can be deduced from this one, requires more preliminary material.

A cut free proof π of a negative formula A is interpreted as a design $\mathcal{D}$ in a localized version of A. So, this is done in two steps. First we choose a localized version of A, say t, we label occurrences of sub-formulæ of A in π by their addresses, and then we construct the design $\mathcal{D}$ by following the rules of π, step by step, from top to bottom. This step by step construction associates a design to each sequent occurring in the proof. Figure 4 summarizes the action of rules on these designs. For each rule, principals formulæ have been annotated by their addresses (between parenthesis). Sometimes these addresses are known to be equal: for instance in the *par* rule the three negative formulæ N, N' and $N \parr N'$ have the same address a. Sequents are annotated by the design constructed so far, drawn from left to right (root on left-hand side, sons right-hand side). In designs, a label, say b, followed by a triangle means a tree of root labeled by b. In the tensor rule, the two triangles means that the two trees of root b have been put side by side and that their roots (both labeled by b) have been identified.

If $\mathcal{D}$ is a design associated to a sequent containing one positive formula P then the unique positive son of the root of $\mathcal{D}$ is labeled by the address of P in the localization of A.

Proposition 14. *The interpretation of a cut-free proof π of a negative formula A is a WMLLP design.*

Theorem 15 (Full completeness). *If t is a localization of a negative formula A and if $\mathcal{D}$ is a WMLLP design in t then there exists a cut-free proof π of A whose interpretation is $\mathcal{D}$.*

$$\frac{}{\vdash 1(b) : [] — a_1^+}\,(\mathit{one}) \qquad \frac{\vdash \Gamma, N(a), N'(a) : \mu + [a,a] — b^+ \lhd}{\vdash \Gamma, N \parr N'(a) : \mu + [a] — b^+ \lhd}\,(\mathit{par})$$

$$\frac{\vdash \Gamma : \mu — b^+ \lhd}{\vdash \Gamma, N(a) : \mu + [a] — b^+ \lhd}\,(\mathit{weak.}) \qquad \frac{\vdash \mathcal{N}, P(a \cdot i) : \mu — a \cdot i^+ \lhd}{\vdash \mathcal{N}, ?P(a) : \mu + [a] — a \cdot i^+ \lhd}\,(\mathit{der.})$$

$$\frac{\vdash \Gamma, P(b) : \ \mu — b^+ \lhd \qquad \vdash \Delta, P'(b) : \nu — b^+ \lhd}{\vdash \Gamma, \Delta, P \otimes P'(b) : \mu + \nu — b^+ \genfrac{}{}{0pt}{}{\lhd}{\lhd}}\,(\mathit{tens.})$$

$$\frac{\vdash \mathcal{N}, N(a \cdot i) : \mu + [a \cdot i] — b^+ \lhd}{\vdash \mathcal{N}, !N(a) : \mu — a^+ — a \cdot i^- — b^+ \lhd}\,(\mathit{prom.})$$

Fig. 4. Interpretation of rules

The demonstration is by induction on $\mathcal{D}$: *one*, tensor and promotion rules give the tree structure of $\mathcal{D}$ (inductively, from root to leaves) and the others rules deal with the labeling of the root. Some nodes of A are not in $\mathcal{D}$: we use weakening rules to add the corresponding sub-formulæ in the proof.

Definition 16. *A pre-play in a WMLLP design $\mathcal{D}$ is an even rooted subtree of $\mathcal{D}$. We use $pp(\mathcal{D})$ to denote the set of pre-plays of $\mathcal{D}$.*

A WMLLP designs $\mathcal{D}$ is equal to the superposition of its pre-plays.

We further identify the canonical localization of a negative formula with its polarized tree.

4.2 Interactions

We have only explained how to interpret each cut-free proof of a negative formula by a design. For others sequents it suffices to complete proofs by use of *par* and promotion rule. Cut rules may be interpreted by using a kind of composition of designs as defined in [Gir01] and further detailed in [Fag02] and [FH02]. Designs seen as a semantics (dynamic in nature) will then satisfy soundness. Composition of designs and interpretation of cut rules in designs is out of the scope of this paper. Anyway, in our framework, one can make programs interacts (*i.e.* compose agents) indifferently in polarized games semantics or in the hypercoherences semantics and benefit of both a dynamic composition (more detailed) and a static composition (simpler and more direct).

4.3 Designs and strategies

Proposition 17. *If π is a cut free proof of a negative formula A, if $\mathcal{D}$ is the WMLLP design interpreting π for the canonical localization of A, and if σ is the* *strategy interpreting π in the polarized game semantics then: $pp(\mathcal{D})$ is equal to $D^-(\sigma) = \{d^-(s) \mid s \in \sigma\}$ and σ is the set whose each element is a negative total order s on a pre-play t of $\mathcal{D}$ such that $<_s^+ = <_t^+$.*

This proposition expresses, in our setting, the correspondence between designs and strategies presented in [FH02].

4.4 Designs and cliques

Proposition 18. *If x is the set interpreting π in the relational model then x is equal to $D^+(pp(\mathcal{D})) = \{d^+(t) \mid t \in pp(\mathcal{D})\}$ where d^+ maps each pre-play to its set of labels ordered by the prefix ordering. Moreover x uniquely determines $\mathcal{D}$.*

The demonstration is by induction on proofs. The trees order on $pp(\mathcal{D})$ is mapped on the trees order on x by d^+ and if the precedence relations for these two orders relate two trees then the bigger is the smaller plus two more nodes. We use this to prove that x uniquely determines $\mathcal{D}$ and to prove the following corollary.

Corollary 19. *The relational interpretation of a proof is a finite set closed for the intersection and superposition of rooted subtrees, that is a finite lattice.*

5 Cliques of chronicles

In ludics even length branches of designs are called *chronicles* and a design is just the prefix tree of its chronicles. In this section, we use chronicles to define a new static semantics of WMLLP enjoying full completeness.

Definition 20. *A* pre-chronicle *in a negative formula A is a balanced rooted subtree p of $t(A)$ such that each positive node has at most one (negative) son (we say that this tree is free of negative branching).*

Proposition 21 (Pre-chronicle means desequentialized chronicle). *Let A be a negative formula. The operation d^+ maps a chronicle of a WMLLP design of type A to a pre-chronicle in A. Conversely, if p is a pre-chronicle in A then p has a unique positive leaf and there exists a totalization s of the order of p alternating between p^- and p^+ and such that $<_s^- = <_p^-$.*

The proof of the second part is by induction and use a simple combinatoric argument to state that p has exactly one positive leaf. The first part is direct.

Definition 22. *A set x of pre-chronicles is said to be* balanced *if each intersection of elements of x is a pre-chronicles. A set x of pre-chronicles is said to be* closed *if each intersection of elements of x is in x. A closed set of pre-chronicles is said to be* saturated *if its least element has exactly two nodes and if each time a pre-chronicle q is a direct successor of a pre-chronicle p then $\sharp q = \sharp p + 2$.*

We restrict the relational model of WMLLP to points which are pre-chronicles. The web of formulæ is defined following this restriction. The interpretation of rules is unchanged. For a tensor rule we just restrict the set of points interpreting the proof to the new web. And to interpret cut rules, if x and y are the sets respectively associated with the first and the second premisses of a cut rule then we form the closure by superposition of x and y, we compose them and we restrict the resulting set to pre-chronicles. The soundness of this semantics is a direct consequence of the following proposition which is itself a direct consequence of proposition 18. We call chronicle relational model the semantics of WMLLP we obtain.

Proposition 23. *If x is the interpretation of a proof in the relational model and x' is the interpretation of the same proof in the chronicle relational model then x is the closure of x' by superposition of rooted subtrees.*

Since the chronicle relational model just consists of a restriction of the webs, we also have (by restriction of graphs and hypergraphs to subsets of vertices) a chronicle hypercoherence semantics and a chronicle coherence space semantics.

Theorem 24 (Full completeness for chronicles hypercoherences). *If x is a balanced clique of negative type A in the chronicle hypercoherence semantics then there exists a proof π of a negative formula B such that x is a subset of the interpretation y of π in the chronicle hypercoherence semantics, $t(B) \subseteq t(A)$ and $\cup x = \cup y$. Moreover if $\cup x$ is total in A then B can be chosen equal to A and if furthermore x is saturated then x is equal to y.*

The closure of x by intersections equipped with the rooted subtrees order is a tree. The theorem is proved by induction on the structure of this tree (from root to leaves) by using the fact that if two pre-chronicles $p \subsetneq q$ are coherent then q contains a son of the unique positive leaf of p (consequence of proposition 8).

Corollary 25 (Full completeness for chronicles coherence spaces). *The same theorem is true for closed cliques in the chronicles coherence semantics.*

6 Futur works

Extension to MELLP. The factorization of D through D^- and D^+ extends to MELLP, but we do not have a satisfactory notion of designs for MELLP. Contraction rules are interpreted as *par* rules but contractions introduce repetitions of nodes in what would be MELLP designs and this makes the situation more complicated. A better handling of localizations in exponentials will be helpful, at least for designing a fully complete static semantics. We currently examine localizations inspired from the *indexed linear logic* of Bucciarelli-Ehrhard [BE01]. This might also help in finding satisfactory exponentials for ludics.

The injectivity of the relational model for multiplicative exponential linear logic is already a conjecture (see [Tor03]). We think possible to prove this conjecture for MELLP in our framework.

In ludics, one adopts a proof-search point of view on proofs: proofs are partially typed and drawn from the conclusion. Our unifying framework introduces this point of view in static semantics. Moreover everything can be defined in a untyped infinitely branching, infinite tree: the *universal arena*, in games semantics. We are interested in applications of coherence spaces and hypercoherences semantics in proof-search areas and also to calculi for which the universal arena is particularly relevant (untyped calculi, for instance).

References

[AJM94] Samson Abramsky, Radha Jagadeesan, and Pasquale Malacaria. Full abstraction for PCF. In *Theoretical Aspects of Computer Software*, pages 1–15, 1994.

[BE01] Antonio Bucciarelli and Thomas Ehrhard. On phase semantics and denotational semantics: the exponentials. *Annals of Pure and Applied Logic*, 109(3):205–241, 2001.

[BHS04] Rick Blute, Masahiro Hamano, and Philip Scott. Softness of hypercoherences and mall full completness. To appear in APAL, 2004.

[Bou04] Pierre Boudes. Projecting games on hypercoherences. Accepted at ICALP, 2004.

[Bra02] J. Bradfield, editor. volume 2471 of *Lecture Notes in Computer Science*, Edinburgh, Scotland, September 2002. European Association for Computer Science Logic, Springer-Verlag.

[Ehr93] Thomas Ehrhard. Hypercoherences: a strongly stable model of linear logic. *Mathematical Structures in Computer Science*, 3, 1993.

[Ehr96] Thomas Ehrhard. Projecting sequential algorithms on strongly stable functions. *Annals of Pure and Applied Logic*, 77, 1996.

[Fag02] Claudia Faggian. Travelling on designs: Ludics dynamics. In Bradfield [Bra02].

[FH02] Claudia Faggian and Martin Hyland. Designs, disputes and strategies. In Bradfield [Bra02], pages 442–458.

[Gir87] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[Gir01] Jean-Yves Girard. Locus Solum. *Mathematical Structures in Computer Science*, 11(3):301–506, June 2001.

[HO00] Martin Hyland and Luke Ong. On full abstraction for PCF: I, II and III. *Information and Computation*, 163(2):285–408, 2000.

[Lau02] Olivier Laurent. Polarized games (extended abstract). In *Proceedings of the seventeenth annual IEEE symposium on Logic In Computer Science*, pages 265–274. IEEE Computer Society Press, July 2002.

[Lau03] Olivier Laurent. Syntax vs. semantics: a polarized approach. Prépublication électronique PPS//03/04//n°17 (pp), Laboratoire Preuves, Programmes et Systèmes, March 2003. Submitted for publication.

[Mel03] Paul-André Melliès. Sequential algorithms and strongly stable functions. To appear in the special issue of TCS: Game Theory Meets Theoretical Computer Science, 2003.

[Tan97] Audrey M. Tan. *Full completeness for models of linear logic.* PhD thesis, University of Cambridge, 1997.

[Tor03] Lorenzo Tortora de Falco. Obsessional experiments for linear logic proof-nets. *Mathematical Structures in Computer Science*, 13(6):799–855, 2003.