On the meaning of focalization

Michele Basaldella Alexis Saurin Kazushige Terui

June 30, 2009

Abstract

In this paper, we use Girard's Ludics to analyze focalization, a fundamental property of linear logic. In particular, we show how this can be realized interactively thanks to section-retraction pairs $(\mathsf{u}_{\alpha\beta}, \mathsf{f}_{\alpha\beta})$ between behaviours $\overline{\alpha}\langle\uparrow(\overline{\beta}\langle\vec{\mathbf{Y}}\rangle), \vec{\mathbf{X}}\rangle$ and $\overline{\alpha\beta}\langle\vec{\mathbf{Y}}, \vec{\mathbf{X}}\rangle$.

1 Introduction

Focalization is a deep outcome of linear logic proof theory, putting to the foreground the role of polarity in logic. It resulted in important advances in various fields ranging from proof-search (the original motivation for Andreoli's study [1] of focalization) and the ability to define synthetic connectives and hypersequentialized calculi [10, 11] to game semantical analysis of logic.

In particular, Focalization deeply influenced Girard's Ludics [12] which is a pre-logical framework which aims to analyze various logical and computational phenomena at a foundational level. For instance, the concluding results of Locus Solum are a full completeness theorem with respect to focalized multiplicative–additive linear logic (MALL). Another characteristics of ludics is that types are built from untyped proofs (called *designs*). More specifically, types (called *behaviours*) are sets of designs closed under a certain closure operation. This view of types as sets of proofs opens a new possibility to discuss focalization and other properties of proofs *at the level of types*.

The purpose of this abstract is to show that Ludics is suitable for analyzing Focalization and that this interactive analysis of Focalization is fruitful. In particular, our study of Focalization in Ludics was primarily motivated by the concluding remarks of the third author's paper on Computational Ludics [17] where focalization on *data designs* was conjectured to correspond to the tape compression theorem of Turing Machines.

Still, for the very reason that Ludics abstracts over Focalization (being built on hypersequentialized calculi) it is not clear whether an analysis of Focalization can (or shall) be pursued in Ludics: an obstacle is, however, that ludics is already fully focalized, so that there seems not to be room to discuss and prove focalization internally. This can be settled by using a dummy shift operator. For instance, a compound formula $L \oplus (M \otimes N)$ of linear logic can be expressed in ludics in two ways; either as a flat behaviour $\oplus\otimes(L, M, N)$ built by a single synthetic connective $\oplus\otimes$ from three subbehaviours L, M, N, or as a compound behaviour $L\oplus\uparrow(M \otimes N)$, which consists of three layers: $M \otimes N$ (positive), $\uparrow(M \otimes N)$ (negative), and $L\oplus\uparrow(M \otimes N)$ (positive).

Focalization can then be expressed as a mapping from the latter to the former behaviour. Hence we can deal with it *as if it were an algebraic law*, which may be compared with other logical isomorphisms such as associativity, distributivity, etc. To be precise, however, focalization is not an isomorphism but is an assymmetric relation. In this paper, we think of it as a *retraction* $L \oplus \uparrow (M \otimes N) \longrightarrow \oplus\otimes(L, M, N)$ which comes equipped with a section $\oplus\otimes(L, M, N) \longrightarrow L \oplus \uparrow (M \otimes N)$.

The aim of our current work is to promote this "algebraic" view of focalization in the setting of ludics. Furthermore, the retraction-section pair can be naturally lifted by applications of logical connectives (Theorem 4.4). Hence we also have focalization inside a compound behaviour (or inside a context). This would allow us to recover the original focalization theorem as a corollary to our "algebraic" focalization, though we leave it as future work.

2 Focalization in linear logic

Linear logic comes from a careful analysis of structural rules in sequent calculus resulting in a very structured proof theory, in particular regarding dualities. A fundamental outcome of those dualities is Andreoli's discovery [1] of focalization, providing the first analysis of polarities in linear logic. Andreoli's contribution lies mainly in the splitting of logical connectives in two groups – positive ($\otimes, \oplus, \mathbf{0}, \mathbf{1}, \exists, !$) and negative ($⅋, \&, \top, \bot, \forall, ?$) connectives.

The underlying meaning of this distinction comes from proof-search motivations. The introduction rules for negative connectives $⅋, \&, \top, \bot, \forall$ are *reversible*: in the bottom–up reading, the rule is deterministic, i.e., there is no choice to make and provability of the conclusion implies provability of the premisses. On the other hand, the introduction rules for positive connectives involve choices: e.g., splitting the context in $\otimes$ rule, or choosing between $\oplus_L$ and $\oplus_R$ rules, resulting in the possibility to make erroneous choices during proof-search. Still, *positive connectives* satisfy a strong property called *focalization*[1]: let us consider a sequent $\vdash F_0, \ldots, F_n$ containing no negative formulas, then there is (at least) one formula F_i which can be used as a *focus* for the search by hereditarily selecting F_i and its positive subformulas as principal formulas up to the first negative subformulas.

This property induces the following strategy of proof–search called *focalization discipline*:

Sequent Γ contains a negative formula	Sequent Γ contains no negative formula
choose any negative formula (e.g. the leftmost one) and decompose it using the only possible negative rule	choose some positive formula and decompose it (and its subformulas) hereditarily until we get to atoms or negative subformulas

A sequent calculus proof is called *focussing* if it respects the focalization discipline. It is proven in [1] that if a sequent is provable, then it is provable with a focussing proof: the focalization discipline is therefore a complete proof-search strategy. Other approaches to focalization consider proof transformation techniques [15, 16]

A very important consequence of focalization is the possibility to consider *synthetic connectives* [11, 4]: a synthetic connective is a maximal cluster of connectives of the same polarity. They are built modulo commutativity and associativity of binary connectives and some syntactical isomorphism [13] of linear logic. For multiplicative— additive—linear logic (**MALL**) the underlying syntactical isomorphism in action is the *distributivity* of $\otimes$ with respect to $\oplus$, namely $(A \oplus B) \otimes C \cong (A \otimes C) \oplus (B \otimes C)$ and its dual.

3 Ludics in three pages

Syntax. We recall the term syntax for designs introduced in [17] which uses a process calculus notation inspired by the close relationship between ludics and linear π-calculus [8].

Designs are built over a given *signature* $\mathcal{A} = (A, \mathsf{ar})$, where A is a set of *names* $a, b, c, \ldots$ and $\mathsf{ar} : A \longrightarrow \mathbb{N}$ assigns an *arity* $\mathsf{ar}(a)$ to each name a. Let $\mathcal{V}$ be a countable set of variables

$\mathcal{V} = \{x, y, z, \dots\}$. Over a fixed signature $\mathcal{A}$, a (proper) *positive action* is $\overline{a}$ with $a \in A$, and a (proper) *negative action* is $a(x_1, \dots, x_n)$ where $x_1, \dots, x_n$ are distinct variables and $\mathsf{ar}(a) = n$. In the sequel, an expression of the form $a(\vec{x})$ always stands for a negative action.

The positive (resp. negative) *designs* P (resp. N) are coinductively generated by the following grammar (where $\mathsf{ar}(a) = n$ and $\vec{x} = x_1, \dots, x_n$):

$$P ::= \Omega \;\Big|\; \maltese \;\Big|\; N_0|\overline{a}\langle N_1, \dots, N_n\rangle, \qquad N ::= x \;\Big|\; \sum a(\vec{x}).P_a,$$

Designs may be considered as infinitary λ-terms with *named* applications and *superimposed* abstractions. $P, Q, \dots$ (resp. $N, M, \dots$, resp. $D, E, \dots$) denote positive (resp. negative, resp. arbitrary) designs. Any subterm E of D is called a *subdesign* of D. Ω is used to encode partial sums: given a set $\alpha = \{a(\vec{x}), b(\vec{y}), \dots\}$ of negative actions, we write $a(\vec{x}).P_a + b(\vec{y}).P_b + \cdots$ to denote the negative design $\sum_\alpha a(\vec{x}).R_a$, where $R_a = P_a$ if $a(\vec{x}) \in \alpha$, and $R_a = \Omega$ otherwise.

A design D may contain free and bound variables. An occurrence of subterm $a(\vec{x}).P_a$ *binds* the free-variables $\vec{x}$ in P_a. Variables which are not under the scope of the binder $a(\vec{x})$ are *free*. $\mathsf{fv}(D)$ denotes the set of free variables occurring in D. Designs are considered *up to α-equivalence*, that is up to renaming of bound variables (see [17] for further details).

A positive design which is neither Ω nor $\maltese$ is either of the form $(\sum a(\vec{x}).P_a)|\overline{a}\langle N_1, \dots, N_n\rangle$ and called a *cut* or of the form $x|\overline{a}\langle N_1, \dots, N_n\rangle$ and called a *head normal form*. The *head variable* x in the design above plays the same role as a pointer in a strategy does in Hyland-Ong games and an address (or locus) in Girard's ludics. On the other hand, a variable x occurring in a bracket (as in $N_0|\overline{a}\langle N_1, \dots, N_{i-1}, x, N_{i+1}, \dots, N_n\rangle$) does not correspond to a pointer nor address but rather to an identity axiom (initial sequent) in sequent calculus, and for this reason is called an *identity*.

A design D is said: *total*, if $D \neq \Omega$; *linear* (or *affine*), if for any subdesign of the form $N_0|\overline{a}\langle N_1, \dots, N_n\rangle$, the sets $\mathsf{fv}(N_0)$, $\dots$, $\mathsf{fv}(N_n)$ are pairwise disjoint.

Normalization. The *reduction relation* $\longrightarrow$ is defined on positive designs as follows:

$$\left(\sum a(x_1, \dots, x_n).P_a\right)|\overline{a}\langle N_1, \dots, N_n\rangle \longrightarrow P_a[N_1/x_1, \dots, N_n/x_n].$$

We denote by $\longrightarrow^*$ its transitive closure. Given two positive designs P, Q, we write $P \Downarrow Q$ if $P \longrightarrow^* Q$ and Q is neither a cut nor Ω. We write $P \Uparrow$ if there is no Q such that $P \Downarrow Q$.

The *normal form function* $[\![\;]\!] : \mathcal{D} \longrightarrow \mathcal{D}$ is defined by corecursion as follows:

$$\begin{array}{rcll} [\![P]\!] & = & \maltese & \text{if } P \Downarrow \maltese; \\ & = & x|\overline{a}\langle [\![\vec{N}]\!]\rangle & \text{if } P \Downarrow x|\overline{a}\langle \vec{N}\rangle; \\ & = & \Omega & \text{if } P \Uparrow; \\ [\![\sum a(\vec{x}).P_a]\!] & = & \sum a(\vec{x}).[\![P_a]\!]; & [\![x]\!] = x. \end{array}$$

A fundamental property of normalization is *associativity*:

$$[\![D[N_1/x_1, \dots, N_n/x_n]]\!] = [\![D]\!][[\![N_1]\!]/x_1, \dots, [\![N_n]\!]/x_n].$$

Orthogonality. In the rest of this work, we restrict ourselves to the special subclass of *total*, *cut-free*, *linear* and *identity-free* designs (corresponding to [12]). Since we work in a cut-free setting, we can simplify our notation: we often identify an expression like $D[N/x]$ with its normal form $[\![D[N/x]]\!]$. Thus, we improperly write $D[N/x] = E$ rather than $[\![D[N/x]]\!] = E$.

A positive design P is *closed* if $\mathsf{fv}(P) = \emptyset$, *atomic* if $\mathsf{fv}(P) \subseteq \{x_0\}$ for a certain fixed variable x_0. A negative design N is *atomic* if $\mathsf{fv}(N) = \emptyset$. Two atomic designs P, N of opposite polarities are said *orthogonal* (written $P \perp N$) when $P[N/x_0] = \maltese$. If $\mathbf{X}$ is a set of atomic designs of the same polarity, then its *orthogonal set* is defined by $\mathbf{X}^\perp := \{E : \forall D \in \mathbf{X},\ D \perp E\}$. Although possible, we do not define orthogonality for nonatomic designs. Accordingly, we only consider *atomic* behaviours which consist of atomic designs.

An (atomic) *behaviour* $\mathbf{X}$ is a set of atomic designs of the same polarity such that $\mathbf{X}^{\perp\perp} = \mathbf{X}$. A behaviour is positive or negative according to the polarity of its designs. We denote positive behaviours by $\mathbf{P}, \mathbf{Q}, \mathbf{R}, \ldots$ and negative behaviours by $\mathbf{N}, \mathbf{M}, \mathbf{K} \ldots$.

There are the least and the greatest behaviours among all positive (resp. negative) behaviours with respect to set inclusion (with $\maltese^- = \sum a(\vec{x}).\maltese$):

$$\mathbf{0}^+ := \{\maltese\}, \qquad \mathbf{0}^- := \{\maltese^-\}, \qquad \top^+ := \mathbf{0}^{-\perp}, \qquad \top^- := \mathbf{0}^{+\perp}.$$

A *positive sequent* $\mathbf{\Gamma}$ is of the form $x_1 : \mathbf{P}_1, \ldots, x_n : \mathbf{P}_n$, where $x_1, \ldots, x_n$ are distinct variables and $\mathbf{P}_1, \ldots, \mathbf{P}_n$ are (atomic) positive behaviours. We denote by $\mathsf{fv}(\mathbf{\Gamma})$ the set $\{x_1, \ldots, x_n\}$. A *negative sequent* $\mathbf{\Gamma}, \mathbf{N}$ is a positive sequent $\mathbf{\Gamma}$ enriched with an (atomic) negative behaviour $\mathbf{N}$, to which no variable is associated. We define:

- $P \models x_1 : \mathbf{P}_1, \ldots, x_n : \mathbf{P}_n$ if $\mathsf{fv}(P) \subseteq \{x_1, \ldots, x_n\}$ and $P[N_1/x_1, \ldots, N_n/x_n] = \maltese$ for any $N_1 \in \mathbf{P}_1^\perp, \ldots, N_n \in \mathbf{P}_n^\perp$.
- $N \models x_1 : \mathbf{P}_1, \ldots, x_n : \mathbf{P}_n, \mathbf{N}$ if $\mathsf{fv}(N) \subseteq \{x_1, \ldots, x_n\}$ and $P[N[N_1/x_1, \ldots, N_n/x_n]/x_0] = \maltese$ for any $N_1 \in \mathbf{P}_1^\perp, \ldots, N_n \in \mathbf{P}_n^\perp$, $P \in \mathbf{N}^\perp$.

Clearly, $N \models \mathbf{N}$ iff $N \in \mathbf{N}$, and $P \models y : \mathbf{P}$ iff $P[x_0/y] \in \mathbf{P}$. Furthermore, associativity implies the following quite useful principle called *closure principle*:

$$P \models \mathbf{\Gamma}, x : \mathbf{P} \Longleftrightarrow \forall N \in \mathbf{P}^\perp, P[N/x] \models \mathbf{\Gamma}, \qquad N \models \mathbf{\Gamma}, \mathbf{N} \Longleftrightarrow \forall P \in \mathbf{N}^\perp, P[N/x_0] \models \mathbf{\Gamma}.$$

Logical connectives and behaviours. We next describe how behaviours are built by means of logical connectives in ludics.

An *n-ary logical connective* α is a pair $(\vec{x}_\alpha, \{a_1(\vec{x}_1), \ldots, a_m(\vec{x}_m)\})$ where $\vec{x}_\alpha = x_1, \ldots, x_n$ is a fixed sequence of variables called the *directory* of α (cf. [12]) and $\{a_1(\vec{x}_1), \ldots, a_m(\vec{x}_m)\}$ is a finite set of negative actions, called the *body* of the connective, such that the names $a_1, \ldots, a_m$ are distinct, the variables $\vec{x}_1, \ldots, \vec{x}_m$ are taken from $\vec{x}_\alpha$ and the order in which the variables occur in $\vec{x}_i$ is the same order in which they occur in $\vec{x}_\alpha$ restricted to $\vec{x}_i$. To enlighten the notation, we often identify a logical connective with its body and so in many occasion we abuse the notation, writing expression like $a(\vec{x}) \in \alpha$. Given a name a, an n-ary logical connective α and behaviours $\mathbf{N}_1, \ldots, \mathbf{N}_n$, $\mathbf{P}_1, \ldots, \mathbf{P}_n$ we define:

$$\overline{\alpha}\langle \mathbf{N}_1, \ldots, \mathbf{N}_n \rangle := \Big(\bigcup_{a(\vec{x}) \in \alpha} \{x_0 | \overline{a}\langle N_{i_1}, \ldots, N_{i_m} \rangle, N_{i_1} \in \mathbf{N}_{i_1}, \ldots, N_{i_m} \in \mathbf{N}_{i_m}\} \Big)^{\perp\perp}$$
$$\alpha(\mathbf{P}_1, \ldots, \mathbf{P}_n) := \overline{\alpha}\langle \mathbf{P}_1^\perp, \ldots, \mathbf{P}_n^\perp \rangle^\perp,$$

where indices $i_1, \ldots, i_m$ are determined by the vector $\vec{x} = x_{i_1}, \ldots, x_{i_m}$ given for each $a(\vec{x}) \in \alpha$.

A behaviour is *logical* if it is inductively built as follows:

$$\mathbf{P} := \overline{\alpha}\langle \mathbf{N}_1, \ldots, \mathbf{N}_n \rangle, \qquad \mathbf{N} := \alpha(\mathbf{P}_1, \ldots, \mathbf{P}_n) \qquad \text{(with } \alpha \text{ an arbitrary logical connective)}$$

Notice that the orthogonal of a logical behaviour is again logical.

Usual **MALL** connectives can be defined as follows ($*$ is a 0-ary name):

$$\begin{array}{llllllllllll} \invamp & := \{\wp(x_1, x_2)\}, & \otimes & := \overline{\invamp}, & \uparrow & := \{\uparrow(x_1)\}, & \perp & := \{*\}, & \bullet & := \overline{\wp}, & \downarrow & := \overline{\uparrow}, \\ \& & := \{\pi_1(x_1), \pi_2(x_2)\}, & \oplus & := \overline{\&}, & \boldsymbol{\downarrow} & := \overline{\boldsymbol{\uparrow}}, & & & \top & := \emptyset, & \iota_i & := \overline{\pi_i}. \end{array}$$

With these logical connectives we can build (semantic versions of) usual linear logic types (we use infix notations such as $\mathbf{N} \otimes \mathbf{M}$ rather than the prefix ones $\otimes\langle \mathbf{N}, \mathbf{M} \rangle$):

$\mathbf{N} \otimes \mathbf{M} \ = \bullet\langle \mathbf{N}, \mathbf{M} \rangle^{\perp\perp}$	$\mathbf{N} \oplus \mathbf{M} \ = (\iota_1\langle \mathbf{N} \rangle \cup \iota_2\langle \mathbf{M} \rangle)^{\perp\perp}$	$\boldsymbol{\downarrow} \mathbf{N} \ = \downarrow\langle \mathbf{N} \rangle^{\perp\perp}$	$\mathbf{0} \ = \emptyset^{\perp\perp}$
$\mathbf{P} \invamp \mathbf{Q} \ = \bullet\langle \mathbf{P}^\perp, \mathbf{Q}^\perp \rangle^\perp$	$\mathbf{P} \& \mathbf{Q} \ = \iota_1\langle \mathbf{P}^\perp \rangle^\perp \cap \iota_2\langle \mathbf{Q}^\perp \rangle^\perp$	$\boldsymbol{\uparrow} \mathbf{P} \ = \downarrow\langle \mathbf{P}^\perp \rangle^\perp$	$\top \ = \emptyset^\perp$

Material and winning designs. Given a behaviour $\mathbf{X}$ and $D \in \mathbf{X}$, there is a "minimal portion" of D which is needed to interact with designs of $\mathbf{X}^{\perp}$. It is called *material part* of D in $\mathbf{X}$. Formally, we define by corecursion the *intersection* $\cap$ on designs as follows:

- $P \cap \Omega = \Omega \cap P = \Omega$;
- $x|\overline{a}\langle \vec{N_i}\rangle \cap x|\overline{a}\langle \vec{M_i}\rangle = x|\overline{a}\langle N_i \vec{\cap} M_i\rangle$ if $N_i \cap M_i$ are defined for every $0 \leq i \leq n$;
- $\sum a(\vec{x}).P_a \cap \sum a(\vec{x}).P'_a = \sum a(\vec{x}).(P_a \cap P'_a)$ if $P_a \cap P'_a$ is defined for every $a \in A$;
- $D \cap E$ is not defined otherwise.

The material part of D in $\mathbf{X}$ is formally defined as: $|D|_{\mathbf{X}} := \bigcap\{E \subseteq D : E \in \mathbf{X}\}$ and is a design of $\mathbf{X}$ [12, 17]. A design $D \in \mathbf{X}$ is said *material* if $D = |D|_{\mathbf{X}}$, *winning* if material and daimon-free. $|\mathbf{X}|$ (resp. $\mathbf{X}_w$) denotes the set of material (resp. winning) designs of $\mathbf{X}$.

Internal completeness. In [12], Girard proposes a purely monistic, local notion of completeness, called *internal completeness*. It means that we can give a precise and direct description to the elements in logical behaviours without using the orthogonality and without referring to any proof system. Logical connectives easily enjoy internal completeness [17]:

- $\overline{\alpha}\langle \mathbf{N}_1, \ldots, \mathbf{N}_n\rangle = \bigcup_{a(\vec{x})\in\alpha} \overline{a}\langle \mathbf{N}_{i_1}, \ldots, \mathbf{N}_{i_m}\rangle \cup \{\maltese\}$.
- $\alpha(\mathbf{P}_1, \ldots, \mathbf{P}_n) = \{\sum a(\vec{x}).P_a : P_a \models x_{i_1} : \mathbf{P}_{i_1}, \ldots, x_{i_m} : \mathbf{P}_{i_m}$ for every $a(\vec{x}) \in \alpha\}$.

In the last equation, P_b can be arbitrary when $b(\vec{x}) \notin \alpha$. For example:

$$\mathbf{P} \,\&\, \mathbf{Q} = \{\pi_1(x_0).P + \pi_2(x_0).Q + \cdots :\ P \in \mathbf{P} \text{ and } Q \in \mathbf{Q}\},$$

where the irrelevant components of the sum are suppressed by "$\cdots$." Up to incarnation (i.e. removal of irrelevant part), $\mathbf{P} \,\&\, \mathbf{Q}$, which has been defined by *intersection*, is isomorphic to the *cartesian product* of $\mathbf{P}$ and $\mathbf{Q}$: a phenomenon called *mystery of incarnation* in [12].

4 An analysis of Focalization in Ludics

Focalized logical behaviours. In the rest of the paper, we shall be interested in how to transform a positive logical behaviour $\mathbf{P} = \overline{\alpha}\langle \uparrow(\overline{\beta}\langle \mathbf{Y}_1, \ldots \mathbf{Y}_m\rangle), \mathbf{X}_2, \ldots, \mathbf{X}_n\rangle$ into a behaviour $\mathbf{Q} = \overline{\alpha\beta}\langle \mathbf{Y}_1, \ldots, \mathbf{Y}_m, \mathbf{X}_2, \ldots, \mathbf{X}_n\rangle$, where $\mathbf{X}_i, \mathbf{Y}_j$ are negative logical behaviours and $\alpha = (\vec{x}_\alpha, \{a_1(\vec{x}_1), a_2(\vec{x}_2), \ldots\})$ with $\vec{x}_\alpha = x_1, \ldots, x_n$, $\beta = (\vec{y}_\beta, \{b_1(\vec{y}_1), b_2(\vec{y}_2), \ldots\}$ with $\vec{y}_\beta = y_1, \ldots, y_m$ such that $\vec{x}_\alpha$ and $\vec{y}_\beta$ are *disjoint*. $\mathbf{Q}$ is called the *focalized behaviour* associated to $\mathbf{P}$ (relative to α, β) while $\alpha\beta$ is the *synthetic connective* associated to α, β.

The choice of having $\uparrow(\overline{\beta}\langle \mathbf{Y}_1, \ldots \mathbf{Y}_m\rangle)$ as $\mathbf{X}_1$ and not, for example as $\mathbf{X}_j$, is of course completely arbitrary and aims at making the presentation simpler. On the other hand, while $\mathbf{X}_2, \ldots, \mathbf{X}_n$ are arbitrary, $\uparrow(\overline{\beta}\langle \mathbf{Y}_1, \ldots \mathbf{Y}_m\rangle)$ has always *this special form*, with the negative connective $\uparrow$ as prefix: focalization roughly asserts that such dummy actions occurring in designs of $\mathbf{P}$ can always be removed by considering synthetic connectives.

In the remaining of this section, and unless otherwise stated, $\mathbf{P}$ and $\mathbf{Q}$ will respectively denote $\overline{\alpha}\langle \uparrow(\overline{\beta}\langle \mathbf{Y}_1, \ldots \mathbf{Y}_m\rangle), \mathbf{X}_2, \ldots, \mathbf{X}_n\rangle$ and $\overline{\alpha\beta}\langle \mathbf{Y}_1, \ldots, \mathbf{Y}_m, \mathbf{X}_2, \ldots, \mathbf{X}_n\rangle$.

Synthetic connectives. In order to define the focalized behaviour $\mathbf{Q}$ we shall define properly the synthetic connective $\alpha\beta$, by specifying its directory and its body:

- The directory of $\alpha\beta$ is $\vec{z}_{\alpha\beta} := y_1, \ldots, y_m, x_2, \ldots, x_n$. Hence, $\alpha\beta$ has arity $n + m - 1$.

- The body of $\alpha\beta$ consists of the set of negative actions $ab(\vec{z})$ defined as follows. First notice that our definition of logical connectives ensures that if some action $a(x_{a_1},\ldots,x_{a_{k_a}})$ in α is such that $x_1 \in x_{a_1},\ldots,x_{a_{k_a}}$, then $x_1 = x_{a_1}$. Thus, for any $a(x_{a_1},\ldots,x_{a_{k_a}})$ in the body of α and $b(y_{b_1},\ldots,y_{b_{l_b}})$ in the body of β, we define a new action ab as:

$$\begin{array}{rll} ab(x_{a_1},\ldots,x_{a_{k_a}}) & \text{if} & x_1 \notin x_{a_1},\ldots,x_{a_{k_a}}, \\ ab(y_{b_1},\ldots,y_{b_{l_b}},x_{a_2},\ldots,x_{a_{k_a}}) & \text{if} & x_1 = x_{a_i}. \end{array}$$

To sum up, we can associate to $\overline{\alpha}\langle\uparrow(\overline{\beta}\langle\mathbf{Y}_1,\ldots\mathbf{Y}_m\rangle),\mathbf{X}_2,\ldots,\mathbf{X}_n\rangle$ its focalized behaviour (relative to α,β) $\overline{\alpha\beta}\langle\mathbf{Y}_1,\ldots\mathbf{Y}_m,\mathbf{X}_2,\ldots,\mathbf{X}_n\rangle$. The following examples illustrate this:

(a) Let $\mathbf{P}$ be $\otimes\langle\uparrow(\downarrow\langle\mathbf{Y}\rangle),\mathbf{X}\rangle$ (written as $\uparrow\downarrow\mathbf{Y}\otimes\mathbf{X}$ in infix notation). Since $\parr$ and $\uparrow$ are respectively $(x_1x_2,\{\wp(x_1,x_2)\})$ and $(y,\{\uparrow(y)\})$ with x_1,x_2,y distinct, we have $\parr\uparrow = (\{yx_2,\wp\uparrow(y,x_2)\})$ and $\mathbf{Q} = \overline{\parr\uparrow}\langle\mathbf{Y},\mathbf{X}\rangle = \otimes\downarrow\langle\mathbf{Y},\mathbf{X}\rangle$. Note that $\otimes\downarrow$ and $\otimes$ are isomorphic.

(b) Let $\mathbf{P}$ be $\oplus\langle\uparrow(\otimes\langle\mathbf{Y}_1,\mathbf{Y}_2\rangle),\mathbf{X}\rangle$ (written $\uparrow(\mathbf{Y}_1\otimes\mathbf{Y}_2)\oplus\mathbf{X}$ in infix notation). Since $\&$ and $\parr$ are respectively $(x_1x_2,\{\pi_1(x_1),\pi_2(x_2)\})$ and $(y_1y_2,\{\wp(y_1,y_2)\})$ we have that $\&\parr = (y_1y_2x_2,\ \{\pi_1\wp(y_1,y_2),\ \pi_2\wp(x_2)\})$ and finally $\mathbf{Q} = \overline{\&\parr}\langle\mathbf{Y}_1,\mathbf{Y}_2,\mathbf{X}\rangle = \oplus\otimes\langle\mathbf{Y}_1,\mathbf{Y}_2,\mathbf{X}\rangle$. Notice that in this case $\pi_2\wp(x_2)$ is just $\pi_2(x_2)$, with an irrelevant change of name.

Now we show how to obtain $\mathbf{Q}$ from $\mathbf{P}$ *interactively*, by means of *interactive functions*.

Interactive functions. Given two positive (resp. negative) logical behaviours $\mathbf{F}$, $\mathbf{G}$, an *interactive function* (i–function for short) $F:\mathbf{F}\longrightarrow\mathbf{G}$ is any design $F\models\mathbf{F}^{\perp},x_0:\mathbf{G}$ (resp. $F\models\mathbf{G},x_0:\mathbf{F}^{\perp}$). We write $F(P)$ for $P[F/x_0]$ if $P\in\mathbf{F}$ (resp. $F(M)$ for $F[M/x_0]$ if $M\in\mathbf{F}$) and F a i–function. We also write $F(\mathbf{F})$ for $\{F(D):D\in\mathbf{F}\}$. Observe that since our setting is fully linear, i–functions have to be intended as "linear" functions. Two examples follow:

(a) A very important i–function is the *fax* [12] (or η-expanded identity) recursively defined as $\mathsf{i}(x_0) := \sum\mathsf{i}(x_0)_a$ with $\mathsf{i}(x_0)_a := a(y_1,\ldots,y_k).x_0|\overline{a}\langle\mathsf{i}(y_1),\ldots\mathsf{i}(y_k)\rangle$.
$\mathsf{i}(x_0)$ plays the role of the identity function for designs: $\mathsf{i}(x_0)(D) = D$ for any D.

(b) We define $\mathsf{u}_{\alpha\beta}:\mathbf{Q}\longrightarrow\mathbf{P}$ as $\mathsf{u}_{\alpha\beta} := \sum_{\alpha\beta}\mathsf{u}_{ab} + \sum_{c\notin\alpha\beta}\mathsf{i}(x_0)_c$ with u_{ab}, for any $ab\in\alpha\beta$, defined as (abbreviating $y_{b_1},\ldots,y_{b_{l_b}}$ by $\mathbf{y}$ and $\mathsf{i}(y_{b_1}),\ldots,\mathsf{i}(y_{b_{l_b}})$ by $\mathsf{i}(\mathbf{y})$):

$$\begin{array}{rcll} \mathsf{u}_{ab} & := & ab(x_{a_1},\ldots,x_{a_{k_a}}).x_0|\overline{a}\langle\mathsf{i}(x_{a_1}),\ldots,\mathsf{i}(x_{a_{k_a}})\rangle & \text{if } x_1\neq x_{a_1} \\ \mathsf{u}_{ab} & := & ab(\mathbf{y},x_{a_2}\ldots,x_{a_{k_a}}).x_0|\overline{a}\langle\uparrow(y).y|\overline{b}\langle\mathsf{i}(\mathbf{y})\rangle,\mathsf{i}(x_{a_2}),\ldots,\mathsf{i}(x_{a_{k_a}})\rangle & \text{if } x_1 = x_{a_1}. \end{array}$$

$\mathsf{u}_{\alpha\beta}$, which sends designs in $\mathbf{Q}$ to designs in $\mathbf{P}$, will be important in analyzing the interactive focalization process of the *focalizing-design* f. The role of $\mathsf{u}_{\alpha\beta}$ is to break a synthetic connective $\alpha\beta$ into its more atomic connectives α and β.

Section-retraction pairs. Given two logical behaviours of the same polarity $\mathbf{F},\mathbf{G}$, a *section-retraction pair* from $\mathbf{G}$ to $\mathbf{F}$ is a pair of i-functions (g,f) with $g:\mathbf{G}\longrightarrow\mathbf{F}$, the section, and $f:\mathbf{F}\longrightarrow\mathbf{G}$, the retraction, such that $f\circ g = \mathsf{i}(x_0)$. A section-retraction pair is *strict* if it sends a daimon-free design to a daimon-free one. Section-retraction pairs can be considered in a context:

Theorem 4.1. *Any (strict) section-retraction pairs between P_i and Q_i (i=1, ..., n) can be extended to a (strict) section-retraction pair between $\alpha(P_1,\ldots,P_n)$ and $\alpha(Q_1,\ldots,Q_n)$ for any logical connective α. The same holds for the positive case.*

Then, Focalization can be expressed as the existence of a section-retraction pair from $\mathbf{Q}$ to $\mathbf{P}$ with $\mathsf{u}_{\alpha\beta}$ as section.

The focalizing-design f. We now introduce the i-function $\mathsf{f}_{\alpha\beta} : \mathbf{P} \longrightarrow \mathbf{Q}$, which will be the retraction associated with $\mathsf{u}_{\alpha\beta}$ and shall interactively build the *focalized designs*. $\mathsf{f}_{\alpha\beta}$ is defined as $\mathsf{f}_{\alpha\beta} := \sum_{\alpha\beta} \mathsf{f}_{ab} + \sum_{c \notin \alpha\beta} \mathsf{i}(x_0)_c$ with, for any $ab \in \alpha\beta$, f_{ab} being defined as:

$$\begin{array}{lcll} \mathsf{f}_{ab} & := & a(x_{a_1}, \ldots, x_{a_{k_a}}).x_0|\overline{ab}\langle \mathsf{i}(x_{a_1}), \ldots, \mathsf{i}(x_{a_{k_a}})\rangle & \text{if } x_1 \neq x_{a_1}, \\ \mathsf{f}_{ab} & := & a(x_1, x_{a_2}, \ldots, x_{a_{k_a}}).x_1|\downarrow\langle \sum_\beta b(\mathbf{y}).x_0|\overline{ab}\langle \mathsf{i}(\mathbf{y}), \mathsf{i}(x_{a_2}), \ldots, \mathsf{i}(x_{a_{k_a}})\rangle\rangle & \text{if } x_1 = x_{a_1}. \end{array}$$

Theorem 4.2. $\mathsf{f}_{\alpha\beta}(\mathbf{P}) = \mathbf{Q}$. *Moreover, winning conditions are preserved:* $\mathsf{f}_{\alpha\beta}(\mathbf{P}_w) \subseteq \mathbf{Q}_w$ *(actually,* $\mathsf{f}_{\alpha\beta}(\mathbf{P}_w) = \mathbf{Q}_w$*).*
$\mathsf{u}_{\alpha\beta}(|\mathbf{Q}|) = |\mathbf{P}|$. *Moreover, winning conditions are preserved:* $\mathsf{u}_{\alpha\beta}(\mathbf{Q}_w) \subseteq \mathbf{P}_w$.

Composing $\mathsf{f}_{\alpha\beta}$ and $\mathsf{u}_{\alpha\beta}$. To establish that $(\mathsf{u}_{\alpha\beta}, \mathsf{f}_{\alpha\beta})$ is a section-retraction pair from $\mathbf{Q}$ to $\mathbf{P}$, we shall study the *composition* of the two i-functions $\mathsf{f}_{\alpha\beta} \circ \mathsf{u}_{\alpha\beta}$. We have:

Proposition 4.3. $\mathsf{f}_{\alpha\beta} \circ \mathsf{u}_{\alpha\beta} = \mathsf{i}(x_0)$.

Proof. By definition of $\mathsf{f}_{\alpha\beta}$ and $\mathsf{u}_{\alpha\beta}$, it is immediate that

$$\mathsf{f}_{\alpha\beta} \circ \mathsf{u}_{\alpha\beta} = [\![\mathsf{f}_{\alpha\beta}(\mathsf{u}_{\alpha\beta})]\!] = [\![\mathsf{u}_{\alpha\beta}[\mathsf{f}_{\alpha\beta}/x_0]]\!] = [\![\textstyle\sum_{\alpha\beta} \mathsf{u}_{ab}[\sum_{\alpha\beta} \mathsf{f}_{ab}/x_0]]\!] + \sum_{c \notin \alpha\beta} \mathsf{i}(x_0)_c.$$

Moreover, since $[\![\sum_{\alpha\beta} \mathsf{u}_{ab}[\sum_{\alpha\beta} \mathsf{f}_{ab}/x_0]]\!] = \sum_{\alpha\beta} [\![\mathsf{u}_{ab}[\mathsf{f}_{ab}/x_0]]\!]$, the left member of the sum can be further decomposed and we have two cases: if $ab(\vec{z})$ is $ab(x_{a_1}, \ldots, x_{a_{k_a}})$, we have that:

$$\begin{array}{lcl} [\![\mathsf{u}_{ab}[\mathsf{f}_{ab}/x_0]]\!] & = & ab(x_{a_1}, \ldots, x_{a_{k_a}}).[\![\mathsf{f}_{ab}|\overline{a}\langle \mathsf{i}(x_{a_1}), \ldots, \mathsf{i}(x_{a_{k_a}})\rangle]\!] \\ & = & ab(x_{a_1}, \ldots, x_{a_{k_a}}).x_0|\overline{ab}\langle [\![\mathsf{i}(\mathsf{i}(x_{a_1}))]\!], \ldots, [\![\mathsf{i}(\mathsf{i}(x_{a_{k_a}}))]\!]\rangle \\ & = & ab(x_{a_1}, \ldots, x_{a_{k_a}}).x_0|\overline{ab}\langle \mathsf{i}(x_{a_1}), \ldots, \mathsf{i}(x_{a_{k_a}})\rangle \\ & = & \mathsf{i}(x_0)_{ab}. \end{array}$$

Otherwise, if $ab(\vec{z}) = ab(y_{b_1}, \ldots, y_{b_{l_b}}, x_{a_2} \ldots, x_{a_{k_a}})$, writing $\mathbf{x}$ for $x_{a_2}, \ldots, x_{a_{k_a}}$, we have that

$$\begin{array}{lcl} [\![\mathsf{u}_{ab}[\mathsf{f}_{ab}/x_0]]\!] & = & ab(\mathbf{y}, \mathbf{x}).[\![f_{ab}|\overline{a}\langle \uparrow(y).y|\overline{b}\langle \mathsf{i}(\mathbf{y})\rangle, \mathsf{i}(\mathbf{x})\rangle]\!] \\ & = & ab(\mathbf{y}, \mathbf{x}).[\![\big(\uparrow(y).y|\overline{b}\langle \mathsf{i}(\mathbf{y})\rangle\big)|\downarrow\langle \sum_\beta b(\mathbf{y}).x_0|\overline{ab}\langle \mathsf{i}(\mathbf{y}), \mathsf{i}(\mathsf{i}(\mathbf{x}))\rangle\rangle]\!] \\ & = & ab(\mathbf{y}, \mathbf{x}).[\![\big(\sum_\beta b(\mathbf{y}).x_0|\overline{ab}\langle \mathsf{i}(\mathbf{y}), \mathsf{i}(\mathsf{i}(\mathbf{x}))\rangle\big)|\overline{b}\langle \mathsf{i}(\mathbf{y})\rangle]\!] \\ & = & ab(\mathbf{y}, \mathbf{x}).x_0|\overline{ab}\langle [\![\mathsf{i}(\mathsf{i}(\mathbf{y}))]\!], [\![\mathsf{i}(\mathsf{i}(\mathbf{x}))]\!]\rangle \;=\; ab(\mathbf{y}, \mathbf{x}).x_0|\overline{ab}\langle \mathsf{i}(\mathbf{y}), \mathsf{i}(\mathbf{x})\rangle \\ & = & \mathsf{i}(x_0)_{ab}. \end{array}$$

Finally, we have obtained that $[\![\mathsf{f}_{\alpha\beta}(\mathsf{u}_{\alpha\beta})]\!] = \mathsf{i}(x_0)$. □

Focalization theorem. We can now conclude with the focalization theorem:

Theorem 4.4 (Focalization Theorem)**.** *For any logical connectives α and β, there is a strict section-retraction pair from $\overline{\alpha\beta}\langle \mathbf{Y}, \mathbf{X}\rangle$ to $\overline{\alpha}\langle \uparrow(\overline{\beta}\langle \mathbf{Y}\rangle), \mathbf{X}\rangle$ which is the pair $(\mathsf{u}_{\alpha\beta}, \mathsf{f}_{\alpha\beta})$.*

An important thing to notice is that theorem 4.1 applies to $(\mathsf{u}_{\alpha\beta}, \mathsf{f}_{\alpha\beta})$ and that the section-retraction pair is strict. This will allow to carry the building of synthetic connectives inside contexts and to ensure we will obtain proofs through the full completeness. $\mathsf{f}_{\alpha\beta}$ is thus a retraction from $\mathbf{P}$ to $\mathbf{Q}$ which will map proofs to proofs with synthetic connectives. Moreover, $\mathsf{u}_{\alpha\beta} \circ \mathsf{f}_{\alpha\beta} : \mathbf{P} \longrightarrow \mathbf{P}$ is an interactive function from $\mathbf{P}$ to $\mathbf{P}$ which preserves winning conditions: given a proof (winning design), it shall build a focused version of that proof.

5 Conclusion and future works

We have considered in this abstract how Focalizationcan be considered from the point of view of ludics itself. In order to do so, we considered interactive functions which have the ability to make a cluster of two positive logical connectives which are separated by a single trivial $\uparrow$ logical connective (that it to merge them in a single synthetic connective), while preserving winning conditions.

Our present work naturally leads to directions that whe shall develop in future works:

- A natural direction is to obtain a proof of the focalization theorem for MALL by combining the results in the present paper with the full-completeness results of Ludics.
- Extending our results to the case of the exponential [2] seems of interest not only because our current analysis is restricted to the linear case, but also because it might clarify several elements of the proof–theory of the exponentials (and their bipolar behaviour).
- The initial motivation of our work was to find an analogous to the tape compression theorem for Turing Machines. We also plan to develop this line of work in the future.

References

[1] Andreoli, J.-M.: Logic Programming with Focusing Proofs in Linear Logic. J. Log. Comput. 2(3) (1992) 297–347.

[2] Basaldella, M., Faggian, C.: Ludics with Repetitions (Exponentials, Interactive Types and Completeness) In LICS. (2009)

[3] Curien, P.-L.: Abstract Böhm trees. Math. Struct. in Computer Science **8** (1998) 559–591.

[4] Curien, P.-L.: Introduction to linear logic and ludics, part II. (2005)

[5] Curien, P.-L.: Notes on game semantics (2006) Manuscript.

[6] Curien, P.L., Herbelin, H.: Abstract machines for dialogue games. (2007)

[7] Faggian, C.: Travelling on designs. In: CSL. (2002) 427–441.

[8] Faggian, C., Piccolo, M.: Ludics is a model for the finitary linear pi-calculus. In: TLCA. (2007) 148–162.

[9] Girard, J.-Y.: A New Constructive Logic: Classical Logic. MSCS 1(3) (1991) 255–296.

[10] Girard, J.-Y.: On the meaning of logical rules I: syntax vs. semantics. In Berger, U., Schwichtenberg, H., eds.: Computational Logic. Springer-Verlag (1999) 215–272.

[11] Girard, J.-Y.: On the meaning of logical rules II : multiplicatives and additives. Foundation of Secure Computation, Berger and Schwichtenberg edts (2000) 183–212.

[12] Girard, J.-Y.: Locus solum: From the rules of logic to the logic of rules. MSCS **11** (2001) 301–506.

[13] Laurent, O.: Étude de la polarization en logique. PhD thesis, Univ. Aix-Marseille II (2002).

[14] Laurent, O.: Polarized games. Ann. Pure Appl. Logic **130** (2004) 79–123.

[15] Laurent, O.: A proof of the focalization property of Linear Logic. Unpublished note, 2004

[16] Miller, D. and Saurin, A.: From proofs to focused proofs: a modular proof of focalization in Linear Logic In: CSL. (2007) 405–419.

[17] Terui, K.: Computational ludics (2008) To appear in Theor. Comput. Sci.