On the meaning of logical completeness

Michele Basaldella* and Kazushige Terui

Research Institute for Mathematical Sciences, Kyoto University, Kitashirakawa Oiwakecho, Sakyo-ku, Kyoto 606-8502, Japan.
{mbasalde,terui}@kurims.kyoto-u.ac.jp

Abstract. Gödel's completeness theorem is concerned with provability, while Girard's theorem in ludics (as well as full completeness theorems in game semantics) is concerned with proofs. Our purpose is to look for a connection between these two disciplines. Following a previous work [1], we consider an extension of the original ludics with contraction and universal nondeterminism, which play dual roles, in order to capture a polarized fragment of linear logic and thus a constructive variant of classical propositional logic.
We then prove a completeness theorem for proofs in this extended setting: for any behaviour (formula) $\mathbf{A}$ and any design (proof attempt) P, either P is a proof of $\mathbf{A}$ or there is a model M of $\mathbf{A}^{\perp}$ which beats P. Compared with proofs of full completeness in game semantics, ours exhibits a striking similarity with proofs of Gödel's completeness, in that it explicitly constructs a countermodel essentially using König's lemma, proceeds by induction on formulas, and implies an analogue of Löwenheim-Skolem theorem.

1 Introduction

Gödel's completeness theorem (for first-order classical logic) is one of the most important theorems in logic. It is concerned with a duality (in a naive sense) between proofs and models: For every formula $\mathbf{A}$,

$$\textit{either} \quad \exists P(P \vdash \mathbf{A}) \quad \textit{or} \quad \exists M(M \models \neg\mathbf{A}).$$

Here P ranges over the set of proofs, M over the class of models, and $P \vdash \mathbf{A}$ reads "P is a proof of $\mathbf{A}$." One can imagine a debate on a general proposition $\mathbf{A}$, where Player tries to justify $\mathbf{A}$ by giving a proof and Opponent tries to refute it by giving a countermodel. The completeness theorem states that exactly one of them wins. Actually, the theorem gives us far more insights than stated.

Finite proofs vs infinite models: A very crucial point is that proofs are always finite, while models can be of arbitrary cardinality. Completeness thus implies Löwenheim-Skolem and compactness theorems, leading to constructions of various nonstandard models.

* Supported by JSPS Postdoctoral Fellowship for Foreign Researcher grant.

Nondeterministic principles: *Any* proof of Gödel's completeness theorem relies on a strong nondeterministic principle such as König's or Zorn's lemma, in contrast to the trivial completeness theorem with respect to the class of boolean algebras.

Matching of two inductions: *Provability* is defined by induction on proofs, while *truth* by induction on formulas. The two inductions are somehow ascribed to the essence of syntax and semantics, respectively, and the completeness theorem states that they do match.

Unlike the real debate, however, there is no interaction between proofs and models in Gödel's theorem. A more interactive account of completeness is given by Girard's *ludics* ([12, 13]; see [10, 4] for good expositions). Ludics is a variant of game semantics, which has the following prominent features.

Monism: Proofs and models are homogeneous entities, called *designs*.

Existentialism: *Behaviours* (semantic types) are built from designs, in contrast to the ordinary game semantics (e.g., Hyland-Ong [14]) where one begins with the definition of arenas (types) and then proceeds to strategies (proofs).

Normalization as interaction: Designs (hence proofs and models) interact together via normalization. It induces an *orthogonality relation* between designs in such a way that $P \perp M$ holds if the normalization of P applied to M converges. A behaviour $\mathbf{A}$ is defined to be a set of designs which is equivalent to its biorthogonal ($\mathbf{A} = \mathbf{A}^{\perp\perp}$).

In this setting, Girard shows a completeness theorem for proofs [13], which roughly claims that any "winning" design in a behaviour is a proof of it. In view of the interactive definition of behaviour, it can be rephrased as follows: For every (logical) behaviour $\mathbf{A}$ and every (proof-like) design P,

$$\textit{either} \quad P \vdash \mathbf{A} \quad \textit{or} \quad \exists M (M \models \mathbf{A}^{\perp} \textit{ and } M \textit{ beats } P).$$

Here, "$M \models \mathbf{A}^{\perp}$" means $M \in \mathbf{A}^{\perp}$, and "M beats P" means $P \not\perp M$. Hence in case $P \not\vdash \mathbf{A}$, we may conclude $P \notin \mathbf{A}^{\perp\perp} = \mathbf{A}$. Notice that $M \models \mathbf{A}^{\perp}$ no more entails absolute unprovability of $\mathbf{A}$; it is rather relativized to each P and there is a real interaction between proofs and models.

Actually, Girard's original ludics is so limited that it corresponds to a polarized fragment of multiplicative additive linear logic, which is too weak to be a stand-alone logical system. As a consequence, one does not observe an opposition between finite proofs and infinite models, since one can always assume that the countermodel M is finite (related to the finite model property for **MALL** [15]). Indeed, the proof of the above completeness is easy once *internal completeness* (a form of completeness which does not refer to any proof system [13]) for each logical connective has been proved.

In this paper, we employ a term syntax for designs introduced in [19], and extend Girard's ludics with duplication (contraction) and its dual: universal nondeterminism (see [1] and references therein). Although our term approach disregards some interesting locativity-related phenomena (e.g., normalization as merging of orders and different sorts of tensors [13]), our calculus is easier to manipulate and closer to the tradition of λ, $\lambda\mu$, $\overline{\lambda}\mu\tilde{\mu}$, π-calculi and others. Our resulting framework is as strong as a polarized fragment of linear logic with exponentials ([4]; cf. also [16]), which is in turn as strong as a constructive version of classical propositional logic.

We then prove the completeness theorem above in this extended setting. Our proof exhibits a striking similarity with Schütte's proof of Gödel's completeness theorem [18]. Given a (proof-like) design P which is not a proof of $\mathbf{A}$, we explicitly construct a countermodel M in $\mathbf{A}^{\perp}$ which beats P, essentially using König's lemma. Soundness is proved by induction on proofs, while completeness is by induction on types. Thus our theorem gives matching of two inductions. Finally, it implies an analogue of Löwenheim-Skolem theorem, which well illustrates the opposition between finite proofs and infinite models.

In game semantics, one finds a number of similar full completeness results. However, the connection with Gödel's completeness seems less conspicuous than ours. Typically, "winning" strategies in Hyland-Ong games most naturally correspond to *Böhm trees*, which can be *infinite* (cf. [5]). Thus, in contrast to our result, one has to impose finiteness/compactness on strategies in an external and noninteractive way, in order to have a correspondence with *finite* λ-terms. Although this is also the case in [1], we show that such a finiteness assumption is not needed in ludics.

2 Designs

2.1 Syntax

We first recall the term syntax for deterministic designs introduced by the second author [19]. We employ a process calculus notation inspired by the close relationship between ludics and linear π-calculus [11].

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

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

The positive (resp. negative) deterministic designs P (resp. N) are coinductively generated by the following grammar:

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

where $\mathsf{ar}(a) = n$ and $\vec{x} = x_1, \ldots, x_n$. Intuitively, designs may be considered as infinitary λ-terms with *named* applications and *superimposed* abstractions. Specifically, a positive design $N_0|\overline{a}\langle N_1, \ldots, N_n\rangle$ can be thought of as iterated application $N_0 N_1 \cdots N_n$ of name $a \in A$, and $a(\vec{x}).P_a$ as iterated abstraction $\lambda\vec{x}.P_a$ of name $a \in A$. A family $\{a(\vec{x}).P_a\}_{a \in A}$ of abstractions indexed by A is then superimposed to form a negative design $\sum a(\vec{x}).P_a$. Each $a(\vec{x}).P_a$ is called its *component*. The reduction rule for designs conforms to this intuition:

$$(\sum a(\vec{x}).P_a)\,|\overline{b}\langle N_1,\ldots,N_n\rangle \longrightarrow P_b[N_1/x_1,\ldots,N_n/x_n].$$

Namely, when the application is of name b, one picks up component $b(\vec{x}).P_b$ from $\{a(\vec{x}).P_a\}_{a\in A}$ and applies β-reduction. Notice that any *closed* positive design P (i.e., a positive design without free variables) has one of the following forms: ✠, Ω and $(\sum a(\vec{x}).P_a)|\overline{a}\langle N_1,\ldots,N_n\rangle$. The last design reduces to another closed one. Hence P eventually reduces to ✠, or Ω or diverges. By stipulating that the normal form of P in the last case is Ω, we obtain a dichotomy between ✠ and Ω: the normal form of a closed positive design is either ✠ or Ω. As we shall see, this induces an orthogonality relation between positive and negative designs.

We also use Ω to encode partial sums. Given a set $\alpha = \{a(\vec{x}), b(\vec{y}), \ldots\}$ of negative actions, we write $a(\vec{x}).P_a + b(\vec{y}).P_b + \cdots$ to denote the negative design $\sum a(\vec{x}).R_a$, where $R_a = P_a$ if $a(\vec{x}) \in \alpha$, and $R_a = \Omega$ otherwise.

Although [19] mainly deals with linear designs, there is no difficulty in dealing with nonlinear ones. To obtain completeness, however, we also need to incorporate the dual of nonlinearity, that is *universal nondeterminism* [1]. It is reminiscent of differential linear logic [8], which has nondeterministic sum as the dual of contraction; the duality is essential for the separation property [17] (see also [7] for separation of Böhm trees). It is also similar to the situation in Hyland-Ong game semantics [14], where *nonlinear* strategies for Player may contain a play in which Opponent behaves *noninnocently*; Opponent's noninnocence is again essential for full completeness.

Definition 1 (Designs). *For a fixed signature $\mathcal{A}$, a **positive** (resp. **negative**) **design** P (resp. N) is a coinductively defined term given as follows:*

$$\begin{array}{rcll} P & ::= & \Omega \;\big|\; \bigwedge_I Q_i & \textit{(positive designs)} \\ Q_i & ::= & N_0|\overline{a}\langle N_1,\ldots,N_n\rangle & \textit{(predesigns)} \\ N & ::= & x \;\big|\; \sum a(\vec{x}).P_a & \textit{(negative designs)}, \end{array}$$

*where $\bigwedge_I Q_i$ is built from a family $\{Q_i\}_{i\in I}$ of **predesigns** with I an arbitrary index set.*

We indicate positive designs by $P, Q, \ldots$, negative designs by $N, M, \ldots$, and arbitrary ones by $D, E, \ldots$. Any subterm E of D is called a *subdesign* of D.

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*. We denote by $\mathsf{fv}(D)$ the set of free variables occurring in D. In analogy with λ-calculus, we always consider designs *up to α-equivalence*, that is up to renaming of bound variables (see [19] for further details). We also identify designs which only differ in indexing: $\bigwedge_I P_i = \bigwedge_J Q_j$ if there is a bijection $\sigma : I \longrightarrow J$ such that $P_i = Q_{\sigma(i)}$ for every $i \in I$.

The *daimon* ✠ is now defined to be the empty conjunction $\bigwedge \emptyset$. A unary conjunction $\bigwedge_{\{i\}} Q_i$ is simply written as Q_i. Furthermore, the conjunction operator can be extended to positive and negative designs: for I, J disjoint sets of indices,

$$\bigwedge_I Q_i \wedge \bigwedge_J Q_i = \bigwedge_{I \cup J} Q_i, \qquad \Omega \wedge P = \Omega,$$
$$\sum a(\vec{x}).P_a \wedge \sum a(\vec{x}).Q_a = \sum a(\vec{x}).(P_a \wedge Q_a), \qquad x \wedge N = \text{undefined}.$$

In particular, we have $P \wedge \maltese = P$ in contrast to [1], which distinguishes the two. By the above convention, conjunction of two positive designs is always defined.

A *cut* is a predesign of the form $(\sum a(\vec{x}).P_a)|\overline{a}\langle N_1, \dots, N_n\rangle$. Otherwise, a predesign is 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 predesign 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. Rather, it corresponds to an identity axiom (initial sequent) in sequent calculus, and for this reason is called an *identity*. If a negative design N simply consists of a variable x, then N is itself 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;
- *deterministic*, if in any occurrence of subdesign $\bigwedge_I Q_i$, I is either empty (and hence $\bigwedge_I Q_i = \maltese$) or a singleton.

Example 1 (Girard's syntax). Girard's original designs [13] can be expressed in our syntax by taking the signature $\mathcal{G} = (\mathcal{P}_{fin}(\mathbb{N}), |\ \ |)$ where $\mathcal{P}_{fin}(\mathbb{N})$ consists of finite subsets of $\mathbb{N}$ and $|\ \ |$ is the function that gives the cardinality to each $I \in \mathcal{P}_{fin}(\mathbb{N})$. Girard's designs correspond to total, deterministic, linear, cut-free and identity-free designs over the signature $\mathcal{G}$. See [19] for more details.

2.2 Normalization

Ludics is an interactive theory. This means that designs, which subsume both proofs and models, interact together via normalization, and types (behaviours) are defined by the induced orthogonality relation. Several ways to normalize designs have been considered in the literature: abstract machines [3, 9, 6, 1], abstract merging of orders [13], and terms reduction [19]. Here we actually extend the last solution. As in pure λ-calculus, normalization is not necessarily terminating, but in our setting a new difficulty arises: the operator $\bigwedge$.

We define the normal forms in two steps, first giving a reduction rule which finds conjunctions of head normal forms whenever possible, and then expanding it corecursively. As usual, let $D[\vec{N}/\vec{x}]$ denote the simultaneous and capture-free substitution of $\vec{N} = N_1, \dots, N_n$ for $\vec{x} = x_1, \dots, x_n$ in D.

Definition 2 (Reduction relation $\longrightarrow$). *The* ***reduction relation*** $\longrightarrow$ *is defined over the set of positive designs as follows:*

$$\Omega \longrightarrow \Omega, \qquad Q \wedge \Big(\sum a(\vec{x}).P_a \mid \overline{b}\langle \vec{N} \rangle \Big) \longrightarrow Q \wedge P_b[\vec{N}/\vec{x}].$$

We denote by $\longrightarrow^*$ the transitive closure of $\longrightarrow$.

Example 2. Let $N = a(x).\maltese \ + \ b(x).(x|\overline{a}\langle x\rangle \wedge x|\overline{b}\langle y\rangle)$. Then:
$N|\overline{a}\langle w\rangle \longrightarrow \maltese, \qquad N|\overline{b}\langle w\rangle \longrightarrow w|\overline{a}\langle w\rangle \ \wedge \ w|\overline{b}\langle y\rangle, \qquad N|\overline{c}\langle w\rangle \longrightarrow \Omega.$

Given two positive designs P, Q, we write $P \Downarrow Q$ and read "P *converges* to Q" if $P \longrightarrow^* Q$ and Q is a conjunction of head normal forms (including the case $Q = \maltese$). We write $P \Uparrow$ and read "P *diverges*" otherwise (typically when $P \longrightarrow^* \Omega$).

Notice that the above reduction relation is completely deterministic. Alternatively, a nondeterministic one can be defined over predesigns and Ω as follows. Given predesigns P_0, R_0 (which can also be seen as positive designs $\bigwedge_{\{0\}} P_0, \bigwedge_{\{0\}} R_0$), we write $P_0 \rightharpoonup R_0$ if $P_0 \longrightarrow Q \wedge R_0$ for some positive design Q. We also write $\Omega \rightharpoonup \Omega$, and $P_0 \rightharpoonup \Omega$ if $P_0 \longrightarrow \Omega$. Then it is easy to see that P_0 converges if and only if *all* nondeterministic reduction sequences from P_0 are finite. Thus our nondeterminism is *universal* rather than *existential.*

Definition 3 (Normal form). *The* ***normal form function*** $[\![\ ]\!] : \mathcal{D} \longrightarrow \mathcal{D}$ *is defined by corecursion as follows:*

$$\begin{aligned} [\![P]\!] &= \textstyle\bigwedge_I x_i|\overline{a}_i\langle [\![\vec{N}_i]\!]\rangle && \text{if } P \Downarrow \textstyle\bigwedge_I x_i|\overline{a}_i\langle \vec{N}_i\rangle; \\ &= \Omega && \text{if } P \Uparrow; \\ [\![\textstyle\sum a(\vec{x}).P_a]\!] &= \textstyle\sum a(\vec{x}).[\![P_a]\!]; && [\![x]\!] = x. \end{aligned}$$

Notice that the dichotomy in the closed case is maintained: for any closed positive design P, $[\![P]\!]$ is either $\maltese$ or Ω.

Theorem 1 (Associativity).
$[\![D[N_1/x_1, \ldots, N_n/x_n]]\!] = [\![D]\!][[\![N_1]\!]/x_1, \ldots, [\![N_n]\!]/x_n]$.

3 Behaviours

3.1 Orthogonality

In the rest of this work, we restrict ourselves to a special subclass of designs: namely, we consider only *total*, *cut-free* and *identity-free* designs. Restriction to identity-free designs is not a serious limitation, since identities can be replaced by suitable infinitary identity-free designs (namely, their infinite η expansions, called *faxes* in [13]). A proof of this fact is given in [19].

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 $[\![D[N/x]]\!] = E$.

Definition 4 (Orthogonality). *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*** *and written $P \bot N$ when $P[N/x_0] = \maltese$. If $\mathbf{X}$ is a set of atomic designs of the same polarity, then its* ***orthogonal set****, denoted by $\mathbf{X}^\bot$, is defined by $\mathbf{X}^\bot := \{E : \forall D \in \mathbf{X},\ D \bot E\}$.*

The meaning of $\bigwedge$ can be clarified in terms of orthogonality. For designs D, E of the same polarity, define $D \preceq E$ iff $\{D\}^\perp \subseteq \{E\}^\perp$. $D \preceq E$ means that E has more chances of convergence than D when interacting with other designs. The following is easy to observe.

Proposition 1. *$\preceq$ is a preorder. Moreover, we have $D \wedge E \preceq D$ and $D \preceq D \wedge D$ for any designs D, E of the same polarity.*

In particular, $P = P \wedge \maltese \preceq \maltese$ for any positive design P. This justifies our identification of $\maltese$ with the empty conjunction $\bigwedge \emptyset$.

Although possible, we do not define orthogonality for nonatomic designs. Accordingly, we only consider *atomic* behaviours which consist of atomic designs.

Definition 5 (Behaviour). *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:

$$\begin{array}{lll} \mathbf{0}^+ := \{\maltese\}, & \top^+ := \mathbf{0}^{-\perp}, & \\ \mathbf{0}^- := \{\maltese^-\}, & \top^- := \mathbf{0}^{+\perp}, & (\maltese^- = \sum a(\vec{x}).\maltese). \end{array}$$

We now introduce the *contexts of behaviours*, which corresponds to *sequents of behaviours* in [13].

Definition 6 (Contexts of behaviours). *A* ***positive context*** $\mathbf{\Gamma}$ *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 context*** $\mathbf{\Gamma}, \mathbf{N}$ *is a positive context* $\mathbf{\Gamma}$ *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 (Theorem 1) implies the following quite useful principle:

Lemma 1 (Closure principle). $P \models \mathbf{\Gamma}, x : \mathbf{P}$ *if and only if* $P[N/x] \models \mathbf{\Gamma}$ *for any* $N \in \mathbf{P}^\perp$*.* $N \models \mathbf{\Gamma}, \mathbf{N}$ *if and only if* $P[N/x_0] \models \mathbf{\Gamma}$ *for any* $P \in \mathbf{N}^\perp$*.*

3.2 Logical connectives

We next describe how behaviours are built by means of logical connectives in ludics. Let us assume that the set of variables $\mathcal{V}$ is equipped with a fixed linear order $x_0, x_1, x_2 \ldots$.

Definition 7 (Logical connectives). *An n-**ary logical connective** α is a finite set of negative actions $a_1(\vec{x}_1), \dots, a_m(\vec{x}_m)$ such that the names $a_1, \dots, a_m$ are distinct and the variables $\vec{x}_1, \dots, \vec{x}_m$ are taken from $\{x_1, \dots, x_n\}$. Given a name a, an n-ary logical connective α and behaviours $\mathbf{N}_1, \dots, \mathbf{N}_n, \mathbf{P}_1, \dots, \mathbf{P}_n$ we define:*

$$\begin{aligned}
&\overline{a}\langle \mathbf{N}_1, \dots, \mathbf{N}_m\rangle := \{x_0|\overline{a}\langle N_1, \dots, N_m\rangle : N_1 \in \mathbf{N}_1, \dots, N_m \in \mathbf{N}_m\},\\
&\overline{\alpha}\langle \mathbf{N}_1, \dots, \mathbf{N}_n\rangle := \left(\bigcup\nolimits_{a(\vec{x})\in\alpha} \overline{a}\langle \mathbf{N}_{i_1}, \dots, \mathbf{N}_{i_m}\rangle\right)^{\perp\perp},\\
&\alpha(\mathbf{P}_1, \dots, \mathbf{P}_n) := \overline{\alpha}\langle \mathbf{P}_1^{\perp}, \dots, \mathbf{P}_n^{\perp}\rangle^{\perp},
\end{aligned}$$

where the indices $i_1, \dots, i_m \in \{1, \dots, n\}$ are determined by the vector $\vec{x} = x_{i_1}, \dots, x_{i_m}$ given for each $a(\vec{x}) \in \alpha$.

In terms of linear logic, the cardinality of the connective α corresponds to the additive arity while the arity of each name to the multiplicative arity.

Example 3 (Linear logic connectives). Usual linear logic connectives can be defined by logical connectives $⅋, \&, \uparrow, \perp, \top$ below; we also give some shorthand notations for readability.

$$\begin{array}{lll}
⅋ := \{\wp(x_1, x_2)\}, & \otimes := \overline{⅋}, & \bullet := \overline{\wp},\\
\& := \{\pi_1(x_1), \pi_2(x_2)\}, & \oplus := \overline{\&}, & \iota_i := \overline{\pi_i},\\
\uparrow := \{\uparrow(x_1)\}, & \downarrow := \overline{\uparrow}, & \downarrow := \overline{\uparrow},\\
\perp := \{*\}, & \top := \emptyset, & (*\ \text{0-ary name}).
\end{array}$$

We do not have exponentials here, because we are working in a nonlinear setting so that they are already incorporated into the connectives. With these logical connectives we can built (semantic versions of) usual linear logic types (we use infix notations such as $\mathbf{N} \otimes \mathbf{M}$ rather than the prefix ones $\otimes\langle \mathbf{N}, \mathbf{M}\rangle$).

$$\begin{array}{ll}
\mathbf{N} \otimes \mathbf{M} = \bullet\langle \mathbf{N}, \mathbf{M}\rangle^{\perp\perp}, & \mathbf{P} ⅋ \mathbf{Q} = \bullet\langle \mathbf{P}^{\perp}, \mathbf{Q}^{\perp}\rangle^{\perp},\\
\mathbf{N} \oplus \mathbf{M} = (\iota_1\langle \mathbf{N}\rangle \cup \iota_2\langle \mathbf{M}\rangle)^{\perp\perp}, & \mathbf{P} \,\&\, \mathbf{Q} = \iota_1\langle \mathbf{P}^{\perp}\rangle^{\perp} \cap \iota_2\langle \mathbf{Q}^{\perp}\rangle^{\perp},\\
\downarrow \mathbf{N} = \downarrow\langle \mathbf{N}\rangle^{\perp\perp}, & \uparrow \mathbf{P} = \downarrow\langle \mathbf{P}^{\perp}\rangle^{\perp},\\
\mathbf{1} = \{x_0|\overline{*}\}^{\perp\perp}, & \perp = \mathbf{1}^{\perp},\\
\mathbf{0} = \emptyset^{\perp\perp}, & \top = \emptyset^{\perp}.
\end{array}$$

The next theorem illustrates a special feature of behaviours defined by logical connectives. It also suggests that nonlinearity and universal nondeterminism play dual roles.

Theorem 2. *Let $\mathbf{P}$ be an arbitrary positive behaviour.*

1. $P \models x_1 : \mathbf{P},\ x_2 : \mathbf{P} \implies P[x_0/x_1, x_0/x_2] \in \mathbf{P}$.
2. $N \wedge M \in \mathbf{P}^{\perp} \implies N \in \mathbf{P}^{\perp}$ *and* $M \in \mathbf{P}^{\perp}$.

*Moreover, if $\mathbf{P} = \overline{\alpha}\langle \vec{\mathbf{N}}\rangle$, the converses of 1. (*duplicability*) and 2.(*closure under $\bigwedge$*) hold.*

Proof. 1. Let $N \in \mathbf{P}^{\perp}$. Then $P[N/x_1, N/x_2] = \maltese$ by assumption.
Hence $P[x_0/x_1, x_0/x_2][N/x_0] = \maltese$, and so $P[x_0/x_1, x_0/x_2] \in \mathbf{P}^{\perp\perp} = \mathbf{P}$.
2. Because of $N \wedge M \preceq N, M$ (Proposition 1).
Closure under $\bigwedge$. Let $N, M \in \mathbf{P}^{\perp} = \overline{\alpha}\langle\vec{\mathbf{N}}\rangle^{\perp}$. To prove $N \wedge M \in \mathbf{P}^{\perp}$, it is sufficient to show that $N \wedge M$ is orthogonal to any $x_0|\overline{a}\langle\vec{K}\rangle \in \bigcup_{a(\vec{x})\in\alpha} \overline{a}\langle\mathbf{N}_{i_1}, \ldots, \mathbf{N}_{i_m}\rangle$. But since x_0 *occurs only once at the head position*, it boils down to $[\![N \wedge M|\overline{a}\langle\vec{K}\rangle]\!] = \maltese$, which is an easy consequence of $[\![N|\overline{a}\langle\vec{K}\rangle]\!] = \maltese$ and $[\![M|\overline{a}\langle\vec{K}\rangle]\!] = \maltese$.
Duplicability. Let $P[x_0/x_1, x_0/x_2] \in \mathbf{P} = \overline{\alpha}\langle\vec{\mathbf{N}}\rangle$. It suffices to show that $P[N/x_1, M/x_2] = \maltese$ holds for any $N, M \in \mathbf{P}^{\perp}$. But we have just proven that $N \wedge M \in \mathbf{P}^{\perp}$, and so $P[x_0/x_1, x_0/x_2][N \wedge M/x_0] = P[N \wedge M/x_1, N \wedge M/x_2] = \maltese$. Since $N \wedge M \preceq N, M$ by Proposition 1, we have $P[N/x_1, M/x_2] = \maltese$. □

Remark 1. Theorem 2 can be considered as an (internal, monistic) form of soundness and completeness for the contraction rule: soundness corresponds to point *1.* whereas completeness to its converse (duplicability).

3.3 Internal completeness

In [13], 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 behaviours (built by logical connectives) without using the orthogonality and without referring to any proof system. Negative logical connectives easily enjoy internal completeness:

Theorem 3 (Internal Completeness (negative case)).
$\alpha(\mathbf{P}_1, \ldots, \mathbf{P}_n) = \{\sum a(\vec{x}).P_a : P_a \models x_{i_1} : \mathbf{P}_{i_1}, \ldots, x_{i_m} : \mathbf{P}_{i_m}$ *for every* $a(\vec{x}) \in \alpha\}$,
where the indices $i_1, \ldots, i_m$ *are determined by the vector* $\vec{x} = x_{i_1}, \ldots, x_{i_m}$.

In the above, P_b can be arbitrary when $b(\vec{x}) \notin \alpha$. Thus our approach is "immaterial" in that we do not consider incarnations and material designs. For example, we have

$$\begin{aligned} \mathbf{P} \,\&\, \mathbf{Q} &= \{\pi_1(x_1).P + \pi_2(x_2).Q + \cdots : \; P \models x_1 : \mathbf{P} \text{ and } Q \models x_2 : \mathbf{Q}\} \\ &= \{\pi_1(x_0).P + \pi_2(x_0).Q + \cdots : \; P \in \mathbf{P} \text{ and } Q \in \mathbf{Q}\}, \end{aligned}$$

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 [13].

As to positive connectives, [13] proves internal completeness theorems for additive and multiplicative ones separately in the linear and deterministic setting. They are integrated in [19] as follows:

Theorem 4 (Internal completeness (linear, positive case)). *When the universe of designs is restricted to linear and deterministic ones, we have*

$$\overline{\alpha}\langle\mathbf{N}_1, \ldots, \mathbf{N}_n\rangle = \bigcup_{a(\vec{x})\in\alpha} \overline{a}\langle\mathbf{N}_{i_1}, \ldots, \mathbf{N}_{i_m}\rangle \cup \{\maltese\}.$$

However, this is no more true with nonlinear designs. A counterexample is given below.

Example 4. Let us consider the behaviour $\mathbf{P} := \downarrow\langle\uparrow(\mathbf{0}^+)\rangle = \downarrow\langle\uparrow(\mathbf{0}^+)\rangle^{\perp\perp}$. By construction, the design $P_0 := x_0|\downarrow\langle\uparrow(x_1).\maltese\rangle$ belongs to $\mathbf{P}$, but then, also any design of the form $P_{n+1} := x_0|\downarrow\langle\uparrow(x_1).P_n\rangle$ belongs to $\mathbf{P}$. Too see this, note that any $N = \sum a(\vec{x}).P_a \in \mathbf{P}^\perp$ has component of the form $\uparrow(y).y|\downarrow\langle M\rangle$ with M arbitrary (more precisely, $\uparrow(y).\bigwedge_I y|\downarrow\langle M_i\rangle$ for some I with M_i arbitrary). Hence we have

$P_{n+1}[N/x_0] = N|\downarrow\langle\uparrow(x_1).P_n[N/x_0]\rangle = (\uparrow(x_1).P_n[N/x_0])\,|\downarrow\langle M\rangle = P_n[N/x_0];$

$P_0[N/x_0] \quad = \maltese.$

This proves $P_{n+1} \in \mathbf{P}$. However, $P_{n+1} \notin \downarrow\langle\uparrow(\mathbf{0}^+)\rangle$, since $\uparrow(x_1).P_n$ is not atomic and so cannot belong to $\uparrow(\mathbf{0}^+)$.

This motivates us to directly prove completeness for proofs, rather than deriving it from internal completeness as in the original work [13]; internal completeness for positives will be further discussed in our subsequent work.

In [1] a weaker form of internal completeness is proved, which is enough to derive a weaker form of full completeness: all *finite* "winning" designs are interpretations of proofs. While such a finiteness assumption is quite common in game semantics, we will show that it can be avoided in ludics.

4 Proof system and completeness for proofs

4.1 Proof system

We will now introduce a proof system. In our system, logical rules are automatically generated by logical connectives. Since the set of logical connectives vary for each signature $\mathcal{A}$, our proof system is parameterized by $\mathcal{A}$. If one chooses $\mathcal{A}$ rich enough, the constant-only fragment of polarized linear logic ([4]; cf. also [16]) can be embedded.

In the sequel, we focus on *logical* behaviours, which are composed by using logical connectives only.

Definition 8 (Logical behaviours). *A behaviour is* ***logical*** *if it is inductively built as follows (α denotes an arbitrary logical connective):*

$$\mathbf{P} := \overline{\alpha}\langle\mathbf{N}_1,\ldots,\mathbf{N}_n\rangle, \qquad \mathbf{N} := \alpha(\mathbf{P}_1,\ldots,\mathbf{P}_n).$$

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

As advocated in the introduction, our monistic framework renders both proofs and models as homogeneous objects: designs.

Definition 9 (Proofs, Models). *A* ***proof*** *is a design in which all the conjunctions are unary. In other words, a proof is a deterministic and $\maltese$-free design. A* ***model*** *is an atomic linear design (in which conjunctions of arbitrary cardinality may occur).*

Given a design D, let $\mathsf{ac}^+(D)$ be the set of occurrences of proper positive actions $\overline{a}$ in D. The *cardinality* of D is defined to be the cardinality of $\mathsf{ac}^+(D)$. Notice that a proof in the above sense can be infinite, so might not "prove" anything. Hence it might be better called a "proof attempt" or "untyped proof."

A *positive* (resp. *negative*) *sequent* is of the form $P \vdash \mathbf{\Gamma}$ (resp. $N \vdash \mathbf{\Gamma}, \mathbf{N}$) where P is a positive proof (resp. N is a negative proof) and $\mathbf{\Gamma}$ is a positive context (see Definition 6) of logical behaviours such that $\mathsf{fv}(P) \subseteq \mathsf{fv}(\mathbf{\Gamma})$ (resp. $\mathsf{fv}(N) \subseteq \mathsf{fv}(\mathbf{\Gamma})$). Intuitively, a sequent $D \vdash \mathbf{\Gamma}$ should be understood as a claim that D is a proof of $\vdash \mathbf{\Gamma}$, or D is of type $\vdash \mathbf{\Gamma}$.

Our proof system consists of three types of inference rules: *positive* $(\overline{\alpha}, \overline{a})$, *negative* (α), and *cut*.

$$\frac{M_1 \vdash \mathbf{\Gamma}, \mathbf{N}_{i_1} \quad \ldots \quad M_m \vdash \mathbf{\Gamma}, \mathbf{N}_{i_m} \quad (z : \overline{\alpha}\langle \mathbf{N}_1, \ldots, \mathbf{N}_n\rangle \in \mathbf{\Gamma})}{z|\overline{a}\langle M_1, \ldots, M_m\rangle \vdash \mathbf{\Gamma}}\ (\overline{\alpha}, \overline{a})$$

$$\frac{\{P_a \vdash \Gamma, \vec{x} : \vec{\mathbf{P}}_a\}_{a(\vec{x}) \in \alpha}}{\sum a(\vec{x}).P_a \vdash \Gamma, \alpha(\mathbf{P}_1, \ldots, \mathbf{P}_n)}\ (\alpha) \qquad \frac{P \vdash \mathbf{\Gamma}, z : \mathbf{P} \quad N \vdash \mathbf{\Delta}, \mathbf{P}^{\perp}}{P[N/z] \vdash \mathbf{\Gamma}, \mathbf{\Delta}}\ (cut)$$

with the proviso:

- In the rule $(\overline{\alpha}, \overline{a})$, $a(\vec{x}) \in \alpha$, $\vec{x} = x_{i_1}, \ldots, x_{i_m}$, and $i_1, \ldots, i_m \in \{1, \ldots, n\}$.
- In (α), $\vec{x} : \vec{\mathbf{P}}_a$ stands for $x_{i_1} : \mathbf{P}_{i_1}, \ldots, x_{i_m} : \mathbf{P}_{i_m}$. A component $b(\vec{y}).P_b$ of $\sum a(\vec{x}_a).P_a$ can be arbitrary when $b(\vec{y}) \notin \alpha$. Hence we again take an "immaterial" approach.

It is also possible to adopt a "material" approach by requiring $P_b = \Omega$ when $b(\vec{y}) \notin \alpha$. Then a proof D is finite (i.e., $\mathsf{ac}^+(D)$ is a finite set) whenever $D \vdash \mathbf{\Gamma}$ is derivable for some $\mathbf{\Gamma}$. Thus, as in ordinary sequent calculi, our proof system accepts only *essentially finite* proofs for derivable sequents (i.e., finite up to removal of irrelevant part).

For linear logic connectives, the positive and negative rules specialize to the following (taking the "material" approach):

$$\frac{M_1 \vdash \mathbf{\Gamma}, \mathbf{N}_1 \quad M_2 \vdash \mathbf{\Gamma}, \mathbf{N}_2 \quad (z : \mathbf{N}_1 \otimes \mathbf{N}_2 \in \mathbf{\Gamma})}{z|\bullet\langle M_1, M_2\rangle \vdash \mathbf{\Gamma}}\ (\otimes, \bullet) \qquad \frac{P \vdash \mathbf{\Gamma}, x_1 : \mathbf{P}_1, x_2 : \mathbf{P}_2}{\wp(x_1, x_2).P \vdash \mathbf{\Gamma}, \mathbf{P}_1 \parr \mathbf{P}_2}\ (\parr)$$

$$\frac{M \vdash \mathbf{\Gamma}, \mathbf{N}_i \quad (z : \mathbf{N}_1 \oplus \mathbf{N}_2 \in \mathbf{\Gamma})}{z|\iota_i\langle M\rangle \vdash \mathbf{\Gamma}}\ (\oplus, \iota_i) \qquad \frac{P_1 \vdash \mathbf{\Gamma}, x_1 : \mathbf{P}_1 \quad P_2 \vdash \mathbf{\Gamma}, x_2 : \mathbf{P}_2}{\pi_1(x_1).P_1 + \pi_2(x_2).P_2 \vdash \mathbf{\Gamma}, \mathbf{P}_1 \,\&\, \mathbf{P}_2}\ (\&)$$

$$\frac{(z : \mathbf{1} \in \mathbf{\Gamma})}{z|\overline{*} \vdash \mathbf{\Gamma}}\ (\mathbf{1}) \qquad \frac{P \vdash \mathbf{\Gamma}}{*.P \vdash \mathbf{\Gamma}, \bot}\ (\bot) \qquad \frac{}{\sum a(\vec{x}).\Omega \vdash \mathbf{\Gamma}, \top}\ (\top)$$

4.2 Completeness for proofs

We now prove soundness and completeness for proofs. In the statement of the theorem below, "$D \vdash \mathbf{\Gamma}$" means that the sequent $D \vdash \mathbf{\Gamma}$ is derivable in our proof system.

Theorem 5 (Soundness). $D \vdash \mathbf{\Gamma} \Longrightarrow D \models \mathbf{\Gamma}$.

Proof. By induction on the derivation of $D \vdash \mathbf{\Gamma}$, using Lemma 1 (the closure principle) and Theorem 2 (1).

Theorem 6 (Completeness for proofs). *For every positive logical behaviour* $\mathbf{P}$ *and every proof* P *(see Definition 9),*

$$P \models x : \mathbf{P} \Longrightarrow P \vdash x : \mathbf{P}.$$

Similarly for the negative case.

The proof below is analogous to Schütte's proof of Gödel's completeness theorem [18], which proceeds as follows:

1. Given an unprovable sequent $\vdash \mathbf{P}$, find an open branch in the cut-free proof search tree.

2. From the open branch, build a countermodel M in which $\mathbf{P}$ is false.

We can naturally adapt 1. to our setting, since the bottom-up cut-free proof search in our proof system is deterministic in the sense that at most one rule applies at each step. Moreover, it never gets stuck at the negative sequent, since a negative rule is always applicable bottom-up.

Suppose now that $P \vdash x : \mathbf{P}$ does not have a derivation. Our goal is to build a model $c(P_x) \in \mathbf{P}^{\perp}$ such that $P \not\perp c(P_x)$.

By König's Lemma, there exists a branch in the cut-free proof search tree,

$$\cfrac{\cfrac{\cfrac{\cfrac{\vdots}{N_1 \vdash \mathbf{\Xi}_1}}{P_1 \vdash \mathbf{\Theta}_1}}{N_0 \vdash \mathbf{\Xi}_0}}{P_0 \vdash \mathbf{\Theta}_0},$$

with $P_0 = P$ and $\mathbf{\Theta}_0 = x : \mathbf{P}$, which is *either* finite and has the topmost sequent $P_{max} \vdash \mathbf{\Theta}_{max}$ with $max \in \mathbb{N}$ to which no rule applies anymore, *or* infinite. In the latter case, we set $max = \infty$. Without loss of generality, we assume that each variable is associated to at most one behaviour. Namely, if $x : \mathbf{P}$ and $x : \mathbf{Q}$ occur in the branch, we have $\mathbf{P} = \mathbf{Q}$ (an assumption needed for Lemma 2 (2)).

We first consider the former case ($max < \infty$) and illustrate how to build a model $c(P_i)$ for $0 \leq i \leq max$ by means of concrete examples. The construction proceeds by downward induction from max to 0.

(i) When $P_{max} = \Omega$, let $c(P_{max}) = \maltese^{-} (= \sum a(\vec{x}).\maltese)$. (ii) Suppose for instance that $P_{max} \vdash \mathbf{\Theta}_{max}$ is of the form $z|\overline{a}\langle\vec{M}\rangle \vdash \mathbf{\Gamma}, z : \mathbf{M} \otimes \mathbf{K}$ but $\overline{a} \neq \bullet$ so that the proof search gets stuck. Then let $c(P_{max}) = \wp(x_l, x_r).\maltese$. (iii) Suppose that we have constructed $c(P_j)$ for $i + 1 \leq j \leq max$, and the relevant part of the branch is of the form:

$$\dfrac{\dfrac{\dfrac{\vdots}{P_{i+1} \vdash \mathbf{\Theta}_{i+1}}}{N_i \vdash \Xi_i}}{\dfrac{P_i \vdash \mathbf{\Theta}_i}{\vdots}} \quad = \quad \dfrac{\dfrac{\dfrac{\vdots}{P_{i+1} \vdash \mathbf{\Gamma}, x : \mathbf{P}, y : \mathbf{Q}}}{\wp(x,y).P_{i+1} \vdash \mathbf{\Gamma}, \mathbf{P} \parr \mathbf{Q}}}{\dfrac{z| \bullet \langle \wp(x,y).P_{i+1}, M\rangle \vdash \mathbf{\Gamma}}{\vdots}},$$

where $\mathbf{\Gamma}$ contains $z : (\mathbf{P} \parr \mathbf{Q}) \otimes \mathbf{M}$. Let:

$c(P_x) = \bigwedge\{c(P_j) : i < j \leq max, P_j$ has head variable $x\}$

$c(P_i) = \wp(x_l, x_r).x_l| \bullet \langle c(P_x), c(P_y)\rangle$.

Here, $c(P_i)$ begins with $\wp(x_l, x_r).x_l$ rather than $\wp(x_l, x_r).x_r$, because the branch goes up to the left direction, choosing the left subformula $\mathbf{P} \parr \mathbf{Q}$. When none of P_j $(i < j \leq max)$ has head variable x, we set $c(P_x) = \maltese^-$.

Next consider the case $max = \infty$. We first define $c_n(P_i)$ for every $n, i < \infty$. Let $c_n(P_i) = \maltese^-$ for $i > n$. For $0 \leq i \leq n$, we build $c_n(P_i)$ by downward induction on i from n to 0, using (iii) above. When $n \to \infty$, each $c_n(P_i)$ grows in the sense that each conjunction $\bigwedge$ obtains more and more conjuncts. This allows us to define $c(P_i)$ for each i by taking the "limit" $\lim_{n\to\infty} c_n(P_i)$, which is roughly speaking the "union" $c(P_i) = \bigcup_{n<\infty} c_n(P_i)$ (cf. [19] for the union of designs). $c(P_x)$ for each variable x is similarly defined. Observe that each $c(P_i)$ and $c(P_x)$ thus constructed are surely models, i.e., atomic linear designs.

Theorem 6 is a direct consequence of the following two lemmas.

Lemma 2. *For $P_i \vdash \mathbf{\Theta}_i$ appearing in the branch, suppose that the head variable of P_i is z and $z : \mathbf{R} \in \mathbf{\Theta}_i$. Then (1) $c(P_i) \in \mathbf{R}^\perp$, and (2) $c(P_z) \in \mathbf{R}^\perp$.*

Proof. By induction on $\mathbf{R}$. (1) When $i = max$ and the case (ii) applies, we have $\wp(x_l, x_r).\maltese \in (\mathbf{M} \otimes \mathbf{K})^\perp$ by internal completeness for negatives (Theorem 3). Suppose that the case (iii) applies to $P_i \vdash \mathbf{\Theta}_i$. Then $c(P_i) = \wp(x_l, x_r).x_l| \bullet \langle c(P_x), c(P_y)\rangle$. By induction hypothesis (2), we have $c(P_x) \in \mathbf{P}^\perp$ and $c(P_y) \in \mathbf{Q}^\perp$. Hence $x_0| \bullet \langle c(P_x), c(P_y)\rangle \in \mathbf{P}^\perp \otimes \mathbf{Q}^\perp = (\mathbf{P} \parr \mathbf{Q})^\perp$. Since x_l, x_r are not free in $c(P_x), c(P_y)$, we have $x_l| \bullet \langle c(P_x), c(P_y)\rangle \models x_l : (\mathbf{P} \parr \mathbf{Q})^\perp, x_r : \mathbf{M}^\perp$. Hence by Theorem 3, $\wp(x_l, x_r).x_l| \bullet \langle c(P_x), c(P_y)\rangle \in (\mathbf{P} \parr \mathbf{Q})^\perp \parr \mathbf{M}^\perp = ((\mathbf{P} \parr \mathbf{Q}) \otimes \mathbf{M})^\perp$. (2) Follows from (1) since $\mathbf{R}^\perp$ is closed under $\bigwedge$ (Theorem 2). □

Lemma 3. *Suppose that the head variable of P_0 is x. Then we have $P_0 \not\perp c(P_x)$.*

Proof. We first prove that there is a nondeterministic reduction sequence

$$P_i[c(P_{v_1})/v_1, \ldots, c(P_{v_m})/v_m] \longrightarrow^* P_{i+1}[c(P_{w_1})/w_1, \ldots, c(P_{w_n})/w_n]$$

for any $i < max$, where $\vec{v}$ and $\vec{w}$ are the free variables of P_i and P_{i+1}, respectively. Suppose that P_i is as in the case (iii) above. By writing $[\theta]$ for $[c(P_{v_1})/v_1, \ldots, c(P_{v_m})/v_m]$ and noting that $c(P_z)$ contains $c(P_i) = \wp(x_l, x_r).x_l| \bullet \langle c(P_x), c(P_y)\rangle$ as conjunct, we have

$$\begin{aligned} P_i[\theta] \;=\; & \; c(P_z)| \bullet \langle \wp(x,y).P_{i+1}[\theta], M[\theta]\rangle \\ \longrightarrow & \; (\wp(x,y).P_{i+1}[\theta])| \bullet \langle c(P_x), c(P_y)\rangle \longrightarrow P_{i+1}[\theta, c(P_x)/x, c(P_y)/y], \end{aligned}$$

as desired. When $max = \infty$, we have obtained an infinite reduction sequence from $P_0[c(P_x)/x]$. Otherwise, we have $P_0[c(P_x)/x] \longrightarrow^* P_{max}[\theta]$. In case (i), $P_{max}[\theta] = \Omega$. In case (ii), $P_{max}[\theta] = c(P_z)|\overline{a}\langle\vec{M}[\theta]\rangle \longrightarrow \Omega$, because $c(P_z)$ contains $c(P_{max}) = \wp(x_l, x_r).\maltese + a(\vec{x}).\Omega + \cdots$ as conjunct. $\square$

This establishes the proof of Theorem 6. Our explicit construction of the model $c(P_x)$ yields a byproduct.

Corollary 1 (Downward Löwenheim-Skolem, Finite model property). *Let P be a proof and $\mathbf{P}$ a logical behaviour. If $P \notin \mathbf{P}$, then there is a countable model $M \in \mathbf{P}^\perp$ (i.e., $\mathsf{ac}^+(M)$ is a countable set) such that $P \not\perp M$.*

Furthermore, when P is linear, there is a finite (and deterministic) model $M \in \mathbf{P}^\perp$ such that $P \not\perp M$.

The last statement is due to the observation that when P is linear the positive rule $(\overline{\alpha}, \overline{a})$ can be replaced with a linear variant:

$$\frac{M_1 \vdash \mathbf{\Gamma}_1, \mathbf{N}_{i_1} \quad \ldots \quad M_m \vdash \mathbf{\Gamma}_m, \mathbf{N}_{i_m}}{z|\overline{a}\langle M_1, \ldots, M_m\rangle \vdash \mathbf{\Gamma}, z : \overline{\alpha}\langle \mathbf{N}_1, \ldots, \mathbf{N}_n\rangle},$$

where $\mathbf{\Gamma}_1, \ldots, \mathbf{\Gamma}_m$ are disjoint subsets of $\mathbf{\Gamma}$. We then immediately see that the proof search tree is always finite, and so is the model $c(P_x)$.

5 Conclusion

We have presented a Gödel-like completeness theorem for proofs in the framework of ludics, aiming at linking completeness theorems for provability with those for proofs. We have explicitly constructed a countermodel against any failed proof attempt, following Schütte's idea based on cut-free proof search. Our proof employs König's lemma and reveals a sharp opposition between finite proofs and infinite models, leading to a clear analogy with Löwenheim-Skolem theorem.

In Hyland-Ong game semantics, Player's "winning" strategies most naturally correspond to possibly infinite Böhm trees (cf. [5]). One could of course impose finiteness/compactness on them to have correspondence with finite proofs. But it would not lead to an explicit construction of Opponent's strategies beating infinite proof attempts. Although finiteness is imposed in [1] too, our current work shows that it is not necessary in ludics.

Our work also highlights the duality:

proof	$\rightleftharpoons$	**model**
deterministic, nonlinear		*nondeterministic, linear*

The principle is that *when proofs admit contraction, models have to be nondeterministic* (whereas do not have to be nonlinear). A similar situation arises, e.g. in [7, 17], when one proves the *separation property* (an analogue of Böhm's theorem [2]), stating that two distinct terms can be distinguished via interaction with a suitable context. Indeed, our construction of countermodels is based on the Böhm out technique that is also crucial for proving separation. To prove the separation property in our setting, however, a more delicate treatment of the conjunction would be required (e.g., D and $D \wedge D$ cannot be separated since $\{D\}^{\perp} = \{D \wedge D\}^{\perp}$).

Acknowledgements. We are deeply indebted to Pierre-Louis Curien, who pointed out a gap in an earlier draft of this paper. Our thanks are also due to the anonymous referees.

References

1. Basaldella, M., Faggian, C.: Ludics with repetition (exponentials, interactive types and completeness). To appear in LICS (2009).
2. Böhm. C.: Alcune proprietà delle forme $\beta - \eta$-normali nel $\lambda - K$-calcolo. Pubblicazioni dell'Istituto per le Applicazioni del Calcolo 696 (1968).
3. Curien, P.-L.: Abstract Böhm trees. Mathematical Structures in Computer Science **8** (1998) 559–591.
4. Curien, P.-L.: Introduction to linear logic and ludics, part II. CoRR **abs/cs/0501039** (2005)
5. Curien, P.-L.: Notes on game semantics (2006) Manuscript.
6. Curien, P.L., Herbelin, H.: Abstract machines for dialogue games. CoRR **abs/0706.2544** (2007)
7. Dezani-Ciancaglini, M., Intrigila, B., Venturini-Zilli, M.: Böhm's theorem for Böhm trees. In : ICTCS'98. (1998) 1–23.
8. Ehrhard, T., Regnier, L.: Differential interaction nets. Theor. Comput. Sci. **364** (2006) 166–195.
9. Faggian, C.: Travelling on designs. In: CSL. (2002) 427–441.
10. Faggian, C.: Interactive observability in ludics: The geometry of tests. Theor. Comput. Sci. **350** (2006) 213–233.
11. Faggian, C., Piccolo, M.: Ludics is a model for the finitary linear pi-calculus. In: TLCA. (2007) 148–162.
12. Girard, J.-Y.: On the meaning of logical rules I: syntax vs. semantics. In Berger, U., Schwichtenberg, H., eds.: Computational Logic. Heidelberg Springer-Verlag (1999) 215 – 272.
13. Girard, J.-Y.: Locus solum: From the rules of logic to the logic of rules. Mathematical Structures in Computer Science **11** (2001) 301–506.
14. Hyland, J.M.E., Ong, C.H.L.: On full abstraction for PCF: I, II, and III. Inf. Comput. **163** (2000) 285–408.
15. Lafont, Y.: The finite model property for various fragments of linear logic. J. Symb. Log. **62** (1997) 1202–1208.
16. Laurent, O.: Polarized games. Ann. Pure Appl. Logic **130** (2004) 79–123.
17. Mazza, D., Pagani, M.: The separation theorem for differential interaction nets. In: LPAR (2007) 393–407.
18. Schütte, K.: Ein System des Verknüpfenden Schliessens. Archiv. Math. Logic Grundlagenf. **2** (1956) 55–67.
19. Terui, K.: Computational ludics (2008) To appear in Theor. Comput. Sci.