Least and Greatest Fixed Points in Ludics

David Baelde, Amina Doumane, Alexis Saurin

Least and Greatest Fixed Points in Ludics

David Baelde[1], Amina Doumane[2], and Alexis Saurin[2]

1 **LSV, ENS Cachan** baelde@lsv.ens-cachan.fr
2 **PPS, CNRS, Université Paris Diderot & Inria**
{amina.doumane,alexis.saurin}@pps.univ-paris-diderot.fr

Abstract

Various logics have been introduced in order to reason over (co)inductive specifications and, through the Curry-Howard correspondence, to study computation over inductive and coinductive data. The logic μMALL is one of those logics, extending multiplicative and additive linear logic with least and greatest fixed point operators.

In this paper, we investigate the semantics of μMALL proofs in (computational) ludics. This framework is built around the notion of design, which can be seen as an analogue of the strategies of game semantics. The infinitary nature of designs makes them particularly well suited for representing computations over infinite data. We provide μMALL with a denotational semantics, interpreting proofs by designs and formulas by particular sets of designs called behaviours. Then we prove a completeness result for the class of "essentially finite designs", which are those designs performing a finite computation followed by a copycat. On the way to completeness, we establish decidability and completeness of semantic inclusion.

1 Introduction

Through the Curry-Howard correspondence, proof theory allows to design and study programming languages in which programs are correct by construction: formulas correspond to types and proofs correspond to well-typed, pure and total programs. Like programs, proofs generally contain irrelevant information which may obfuscate their computational contents, making it hard to tell when two proofs correspond to the same computation, or to characterize the class of computations being expressible as proofs. A major goal of proof theory is to tackle these problems, by identifying and eliminating such syntactic noise to get down to the essence of proofs.

Following this tradition, the proof theory of least and greatest fixed points provides a way to design and study programming constructs associated to inductive and coinductive types. Such types can be encoded using second-order quantification, *e.g.*, $\forall X.\ X \to (X \to X) \to X$ represents natural numbers through primitive recursion. However, the encoding has several undesirable effects: it notably forces impredicativity into the system, and results in indirect ways of computing over (co)inductive objects. These two reasons have motivated the introduction of fixed points in type theory. Mendler [16] and Matthes [14] extended second-order λ-calculus with least and greatest fixed point types and an associated generic primitive recursion operator. Similar developments took place in richer type theories, supporting the introduction of inductive and coinductive specifications in tools such as Coq or Agda. However, these aspects are far from being fully understood: there is a history of bugs pertaining to the guard condition that restricts (co)recursion to ensure totality in these systems. This has spurred the development of other approaches to (co)induction, such as sized types [3] and (co)patterns [1]. The works cited above are mostly concerned with strong normalization, and do not investigate the computational content of proofs beyond this property. Various other works, generally taking place in weaker logics, have investigated more closely the semantics of proof on (co)inductive types, notably by means of infinite proofs and games [18, 10, 7, 8, 6].

Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In this paper, we carry out a similar semantical investigation, revealing the computational content of proofs by means of an infinitary semantics, and tackling the difficult problem of completeness for this semantical interpretation. We work with the system μMALL [2], which extends multiplicative and additive linear logic[1] with least and greatest fixed point operators. In addition to simple fixed point unfolding rules, μMALL features, like most of the systems cited above, generic primitive (co)recursion rules shown below in a simplified two-sided form:

$$\frac{F[S/X] \vdash S}{\mu X.F \vdash S}\ (\mu_L) \qquad \frac{S \vdash F[S/X]}{S \vdash \nu X.F}\ (\nu_R)$$

The induction principle (μ_L) behaves in cut elimination as a recursive process, transforming any derivation of $\vdash \mu X.F$ into a derivation of $\vdash S$. Now, we can express in that framework the type of lists of A as $L = \mu X.1 \oplus (A \otimes X)$ and the type of infinite streams of A as $S = \nu X.A \otimes X$. We can then define a proof (shown next) that concatenates a list and a stream into a stream, by recursing over the list with the invariant $S \multimap S$. It is not trivial to convince oneself that this proof does compute the concantenation function. More generally, it is hard to tell what such proofs compute, when two proofs compute the same function, etc. It requires to step back from the finite, syntactic proof system under consideration and to start considering its semantics; this is the topic of the present paper.

$$\dfrac{\dfrac{}{1 \vdash S \multimap S}\,(\ldots,\mathrm{ax}) \qquad \dfrac{\dfrac{\dfrac{}{A, S \multimap S, S \vdash A \otimes S}\,(\ldots,\mathrm{ax})}{A, S \multimap S, S \vdash S}\,(\mathrm{unfold}_R)}{A \otimes (S \multimap S) \vdash S \multimap S}\,(\otimes_L, \multimap R)}{\dfrac{1 \oplus (A \otimes (S \multimap S)) \vdash S \multimap S}{L \vdash S \multimap S}\,(\mu_L)}\,(\oplus_L)$$

As our domain of interpretation of proofs, we consider ludics [11] which can be regarded as a variant of game semantics, where the basic objects are well-behaved strategies, called designs. Girard introduced ludics with the aim of bringing closer together syntax and semantics in the study of proofs and proved a full completeness result with respect to proofs of a polarized variant of MALL. Extending this interpretation to all of linear logic, including exponentials, has been challenging and required to deal with non-determinism [4]. As we shall see, accounting for least and greatest fixed points is much easier, and can essentially be done in Girard's original framework. Still, we shall work in Terui's reformulation of ludics, *computational ludics* [19], since it is more convenient to work with and slightly more general; for instance the objects of computational ludics may contain cut while Girard's original designs are cut-free: this happens to be very handy when working with greatest fixed point.

Contributions. Our first contribution is a denotational semantics for (a polarized version of) μMALL: we interpret formulas as well-behaved sets of designs, and proofs as designs belonging to the conclusion sequent's interpretation. Our second contribution is a completeness result for a class of designs we call *essentially finite designs* (EFD) which are designs performing a finite computation followed by a copycat. To prove this result, we investigate semantic inclusion, proving its decidability (*i.e.*, whether the intepretations of two formulas are included in each other) and completeness (if semantic inclusion holds, the corresponding entailment is μMALL-provable). This last result relies on a circular proof system, in the style of Santocanale [18], as a stepping stone between infinite designs and finite proofs.

Outline. We introduce μMALLP (the polarized variant of μMALL) in Section 2 followed

[1] While linearity is certainly a restriction, it is not a severe one with respect to our main interest in this paper. Note moreover that μMALL is already very expressive as it contains at least all primitive recursive functions.

by ludics in Section 3. We then define the interpretation of μMALLP proofs in ludics in Section 4, and prove its soundness. Finally, we establish completeness for EFD in Section 5.

2 Linear Logic with Fixed Points

In this section, we formally introduce our logic with least and greatest fixed points. As usual when aiming at ludics interpretation [11, 4, 19] we will actually be working with a polarized version of μMALL [2], μMALLP, to which the present section is dedicated.

We assume two infinite and disjoint sets $\mathcal{V}_P$ and $\mathcal{V}_N$, whose elements are respectively called positive and negative variables and denoted by X_P and X_N, or simply X when their polarity is irrelevant or can be inferred from the context.

▶ **Definition 1.** The sets of *positive preformulas* $P, Q, \ldots$ and of *negative preformulas* $N, M, \ldots$ are inductively defined by the following grammar:

$$\begin{array}{rcl} P,\ Q & ::= & X_P \mid X_N^\perp \mid 1 \mid 0 \mid N \oplus M \mid N \otimes M \mid \downarrow N \mid \mu X_P.P \\ N, M & ::= & X_N \mid X_P^\perp \mid \perp \mid \top \mid P \& Q \mid P \parr Q \mid \uparrow P \mid \nu X_N.N \end{array}$$

The connectives μ and ν are variable binders, and the notions of free and bound variables are as usual. *Formulas* are those preformulas with no free variables. A preformula is said to be *monotonic* if it contains no negated variable $X^\perp$, neither free nor bound. A formula is said to be *degenerate* if it contains either $\mu X.X$ or $\nu X.X$ as a subformula.

Nested fixed points correspond to iterated (co)inductive definitions. For example, $\mathrm{Nat} := \mu X.\ (\uparrow 1) \oplus (\uparrow X)$ is the type of natural numbers, and $\nu Y.\ \uparrow((\uparrow \mathrm{Nat}) \otimes Y)$ is the type of infinite streams of natural numbers. Fixed points can also be interleaved, which corresponds to mutual (co)inductive definitions. For example, $\mu X.\ T \otimes (\nu Y.\ \uparrow((\uparrow 1) \oplus ((\uparrow X) \otimes Y)))$ is the type of arbitrarily branching well-founded trees, with data of type T as every node — such trees have no infinite branch, but each node may have infinitely many children.

Our syntax classifies μ as positive and ν as negative. This is a natural choice but it is not forced: all of this work could be done by taking the opposite classification, which is consistent with the observations made in the study of focusing for μMALL [2]. In a nutshell, the polarity of a fixed point formula is not forced by the fixed point operator but rather by the formula inside the fixed point. In that setting, the formulas $\mu X.X$ and $\nu X.X$ have no meaningful polarity. We shall thus assume from now on that all formulas are non-degenerate.

▶ **Definition 2.** *Negation* is the involutive operation mapping positive to negative preformulas, and vice versa, such that:

$$\begin{array}{c} (F_1 \parr F_2)^\perp = F_1^\perp \otimes F_2^\perp \quad (F_1 \& F_2)^\perp = F_1^\perp \oplus F_2^\perp \quad (\uparrow F)^\perp = \downarrow F^\perp \\ (\nu X.F)^\perp = \mu Y.(F^\perp[Y^\perp/X]) \quad (X_N)^\perp = X_N^\perp \quad (X_P^\perp)^\perp = X_P \quad \top^\perp = 0 \quad \perp^\perp = 1 \end{array}$$

From now on, we restrict our attention to monotonic (pre)formulas. This natural restriction rules out formulas such as $\mu X.\downarrow X^\perp$, which would yield inconsistencies. Assuming monotonicity amounts to fully remove negation from our syntax — the presence of negated variables in it is only useful to be able to define negation. We may still use negation as an operation on formulas, since it preserves monotonicity.

We denote by $F[\vec{G}/\vec{X}]$ the preformula obtained by the simultaneous capture-avoiding substitution of the variables $\vec{X}$ by the preformulas $\vec{G}$. When considering a substitution $F[\vec{G}/\vec{X}]$, we always assume implicitly that the polarities of $\vec{G}$ are adequate to those of $\vec{X}$.

Identity rules:

$$\frac{}{\vdash P, P^\perp}\ (\text{ax}) \qquad \frac{\vdash \Gamma, P^\perp \quad \vdash \Delta, P}{\vdash \Gamma, \Delta}\ (\text{cut})$$

Fixed point rules:

$$\frac{\vdash \Gamma, P[\mu X.P/X]}{\vdash \Gamma, \mu X.P}\ (\mu) \qquad \frac{\vdash \Gamma, S \quad \vdash S^\perp, N[S/X]}{\vdash \Gamma, \nu X.N}\ (\nu)$$

MALL rules:

$$\frac{}{\vdash \Gamma, \top}\ (\top) \qquad \frac{\vdash \Gamma}{\vdash \Gamma, \bot}\ (\bot) \qquad \frac{}{\vdash 1}\ (1) \qquad \frac{\vdash \Gamma, P}{\vdash \Gamma, \uparrow P}\ (\uparrow) \qquad \frac{\vdash \Gamma, N}{\vdash \Gamma, \downarrow N}\ (\downarrow)$$

$$\frac{\vdash \Gamma, P_1 \quad \vdash \Gamma, P_2}{\vdash \Gamma, P_1 \& P_2}\ (\&) \qquad \frac{\vdash \Gamma, N_i}{\vdash \Gamma, N_1 \oplus N_2}\ (\oplus_i) \qquad \frac{\vdash \Gamma, P_1, P_2}{\vdash \Gamma, P_1 \parr P_2}\ (\parr) \qquad \frac{\vdash \Delta, N_1 \quad \vdash \Gamma, N_2}{\vdash \Gamma, \Delta, N_1 \otimes N_2}\ (\otimes)$$

In these rules, Γ and Δ denote positive sequents, *i.e.*, ones that contain only positive formulas.

Figure 1 The μMALLP sequent calculus proof system.

▶ **Definition 3.** The proof system μMALLP is given in Figure 1. It is a focused sequent calculus over our polarized syntax, meaning that its sequents must contain at most one negative formula. A sequent is said to be *negative* when it contains a negative formula, and it is *positive* otherwise. In Figure 1, Γ and Δ always denote positive sequents.

Reading proofs in a proof search (bottom-up) fashion, the polarity restriction on sequents means that negative rules must be applied eagerly, *i.e.*, as soon as the sequent contains a negative formula. This constraint on the shape of proofs is a very mild form of focusing. Note that we can simply translate between μMALL and μMALLP, in the same way as is done between MALL and MALLP: μMALL formulas are translated into μMALLP formulas by inserting shift connectives, and any μMALL proof can be turned into a μMALLP proof of the translated conclusion sequent by inserting shift rules[2]; in the other direction, shifts are simply erased.

As mentionned in the introduction, the fixed point rules of μMALLP can be understood from Knaster-Tarski's characterization of an operator's extremal fixed points in complete lattices. Rule (μ) expresses that $\mu X.P$ is a pre-fixed point of $X \mapsto P$, provided that one reads implication as inclusion $(P[\mu X.P/X] \multimap \mu X.P)$. Similarly, we may express that the greatest fixed point is greater than all post-fixed points by the following rule:

$$\frac{S \vdash N[S/X]}{S \vdash \nu X.N}\ (\nu_0)$$

Rule (ν) of Figure 1 is obtained from this one by combining it with a cut against the co-invariant S — this presentation is preferred because it yields a system that enjoys cut elimination.

The above explanation may be helpful to understand the rules at first, but it does not say anything about their computational interpretation. Cut elimination holds for μMALLP: the cut reduction system given in [2] can straightforwardly be adapted to the polarized setting of μMALLP. As the reader may expect, the only specific case is the one involving least and greatest fixed points. This cut reduction step, shown in Figure 2, relies on the *functoriality* construction: if Π is a proof of a sequent $\vdash P, N$ then $F_G(\Pi)$ is a particular

[2] This translation essentially cancels the focusing constraint of the μMALLP proof system by inserting shifts. A more demanding task would be to establish completeness of the focused μMALLP proof system given here with respect to an unfocused proof system for μMALLP. We do not address this (unrelated) issue, but expect that it would be possible along the lines of the focusing result for μMALL [2].

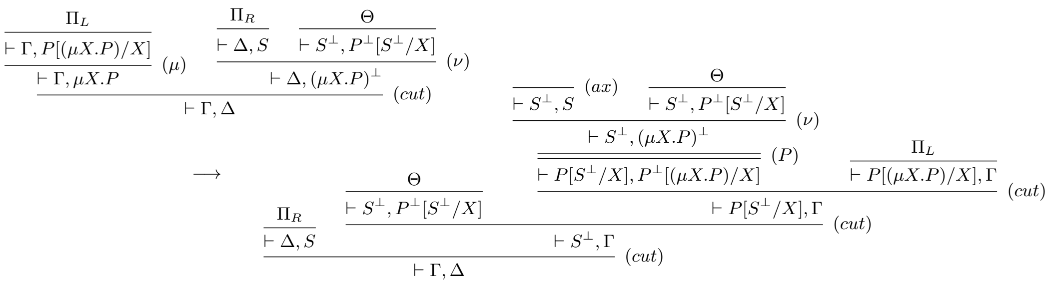

Figure 2 (μ)–(ν) Key cut-elimination step.

proof of $\vdash G[P/X], G^{\perp}[N^{\perp}/X]$, whose precise definition may be found in [2]. Operationally, functoriality should be viewed as a `map` operator in functional programming. The role of $F_G(\Pi)$ is to apply Π to all occurrences of X in G. Roughly, its type may be read as $(N^{\perp} \to P) \to (G[N^{\perp}/X] \to G[P/X])$. In the cut reduction step of Figure 2, this has the effect of propagating the (ν) rule to the next occurrences of $(\mu X.P)^{\perp}$ in $P^{\perp}[(\mu X.P)^{\perp}/X]$. Overall, the reduction achieves in this way the (co)recursive computational behaviour described in the introduction[3]. Intuitively, this process terminates because each application of this reduction consumes a (μ) rule; see [2] for the formal argument.

3 Computational Ludics

Ludics is an interactive framework reminiscent from and somehow intermediate between game semantics [12] and realizability [13]. We recall, in the setting of computational ludics [19], the necessary definitions and properties of

- designs (§ 3.1), which correspond to strategies,
- orthogonality (§ 3.2), which corresponds to interaction, and
- behaviours (§ 3.3), which correspond to arenas or interactive types.

In game semantics, arenas are defined first and strategies are defined as sets of plays (or as sets of views) compatible with these arenas. However, in ludics, arenas are a secondary notion, derived from that of designs since behaviours are obtained from designs by orthogonality; we develop this comparison in § 3.4.

3.1 Designs

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

▶ **Definition 4.** For a fixed signature $\mathcal{A}$, the class of *positive designs* $p, q, \dots$ and *negative designs* $n, m, \dots$ are coinductively defined as follows (with $\mathrm{ar}(a) = \mathrm{card}(\vec{x}_a) = k$):

$$p \;::=\; \Omega \;\mid\; \maltese \;\mid\; (n_0 \mid \overline{a}\langle n_1, \dots, n_k\rangle) \qquad\qquad n \;::=\; x \;\mid\; \sum a(\vec{x}_a).p_a$$

The formal sum $\sum a(\vec{x}_a).p_a$ is the $\mathcal{A}$-indexed family $\{a(\vec{x}_a).p_a\}_{a \in \mathcal{A}}$.

[3] Note that in our classical linear logic setting, induction and coinduction (in other words, recursion and corecursion) become the same.

We write $\sum_{K\subseteq\mathcal{A}} a(\vec{x}_a).p_a$ to denote the *design* $\sum a(\vec{x}_a).q_a$ where $q_a = p_a$ if $a \in K$ and $q_a = \Omega$ otherwise. We denote by Ω^- the design $\sum a(\vec{x}_a).\Omega$. In a negative design $\sum a(\vec{x}_a).p_a$, $a(\vec{x}_a)$ binds the variables $\vec{x}_a$ appearing in p_a. Variables which are not under the scope of a binder are free. The free variables of a design d are denoted by $\mathrm{fv}(d)$. We identify two designs which are α-equivalent, *i.e.*, which are equal up to renaming of their bound variables. We denote by $d[\vec{n}/\vec{x}]$ the design obtained by a simultaneous and capture-free substitution of the variables $\vec{x}$ by the negative designs $\vec{n}$. The reader is referred to [19] for precise definitions.

▶ **Definition 5** (l-designs, standard designs)**.** A design of the form $n_0 \mid \overline{a}\langle n_1,\dots,n_k\rangle$ where n_0 is not a variable is called a *cut*. An occurrence of a variable x is called an *identity* if it occurs as $n_0 \mid \overline{a}\langle n_1,\dots,x,\dots,n_k\rangle$. We call a design *identity-free* (resp. *cut-free*) if it does not contain an identity (resp. a cut) as a subdesign. A design d is called *linear* if for every positive subdesign $n_0 \mid \overline{a}\langle n_1,\dots,n_k\rangle$ of d, the sets $\mathrm{fv}(n_0),\dots,\mathrm{fv}(n_k)$ are pairwise disjoint. An *l-design* is a design d which is linear, identity-free and such that $\mathrm{fv}(d)$ is finite. A *standard* design is a cut-free l-design.

▶ **Definition 6** (MALL signature)**.** In order to interpret polarized MALL proofs, the following signature, and associated notations, are useful: $\mathcal{A} = \{\bot,\uparrow,\&_1,\&_2,\parr\}$ with $\mathrm{ar}(\bot) = 0, \mathrm{ar}(\uparrow) = \mathrm{ar}(\&_1) = \mathrm{ar}(\&_2) = 1, \mathrm{ar}(\parr) = 2$. When considering this signature, we write 1 rather than $\overline{\bot}$, $\downarrow$ rather than $\overline{\uparrow}$, $\oplus_i$ rather than $\overline{\&_i}$ and $\otimes$ rather than $\overline{\parr}$.

We define in the following the design η_F, the infinitary η-expansion of the axiom over F:

▶ **Definition 7.** Let F be a MALL formula. The design η_F is coinductively defined by:

$$\begin{array}{lclcl}
\eta_{F_1\otimes F_2} & = & \eta_{F_1\parr F_2} & = & \parr(x_1,x_2).(x_0 \mid \otimes\langle \eta_{F_1,d}[x_1/x_0], \eta_{F_2}[x_2/x_0]\rangle) \\
\eta_{F_1\oplus F_2} & = & \eta_{F_1\& F_2} & = & \sum_{i=1,2} \&_i(x_i).(x_0 \mid \oplus_i\langle \eta_{F_i}[x_i/x_0]\rangle) \\
\eta_{\downarrow F} & = & \eta_{\uparrow F} & = & \uparrow(x_1).(x_0 \mid \downarrow\langle \eta_F[x_1/x_0]\rangle) \\
\eta_{\sigma Y.F} & = & \eta_{F[\sigma Y.F/Y]} & & \text{for } \sigma\in\{\mu,\nu\}
\end{array}$$

▶ **Example 8.** Here are two additional examples of designs defined on the MALL signature:

$$d_1 = \&_1(x_1).(x_1 \mid 1) + \&_2(x_2).(x_2 \mid \oplus_1\langle \uparrow(y).(y \mid 1)\rangle)$$
$$d_2 = \parr(x_1,x_2).(x_2 \mid \downarrow\langle d_2\rangle)$$

▶ Remark. Designs (on the MALL signature) can be viewed as abstractions of (suitably polarized) MALL proofs. For instance d_1 abstracts the (unique) cut-free proof of $\vdash 1\&(\uparrow 1\oplus\bot)$.

The previous remark is the basis of the usual interpretation of MALL in ludics that we will extend, in the rest of the paper, into an interpretation of μMALLP. But first, as ludics is all about interaction, we turn to cut-elimination and orthogonality.

3.2 Cut-elimination and Orthogonality

Cuts can be reduced by the relation $\to$ defined as follows:

▶ **Definition 9.** The relation $\to$ is defined on positive designs as follows:

$$(\sum a(\vec{x}_a).p_a) \mid \overline{b}\langle\vec{n}\rangle \to p_b[\vec{n}/\vec{x}_b].$$

The reflexive and transitive closure of $\to$ is denoted $\to^\star$. We write $p \Downarrow q$ if $p \to^\star q$ and q is neither a cut, nor the design Ω. If such a design q does not exist, we write $p \Uparrow$. We define $\perp\!\!\!\perp$ to be the least set of positive designs containing $\maltese$ and closed by anti-reduction: $\perp\!\!\!\perp = \{d : d \to^\star \maltese\}$.

To eliminate cuts from designs, we coinductively propagate the relation $\Downarrow$ to subdesigns. The obtained normal form $(\!|d|\!)$ enjoys a weak form of Church-Rosser property.

▶ **Definition 10** (Normal form)**.** The function $(\!|.|\!)$ on designs is coinductively defined by:

$$\begin{array}{lll} (\!|p|\!) = \maltese & if \quad p \Downarrow \maltese; & (\!|x|\!) = x \\ \quad = \Omega & if \quad p \Uparrow; & (\!|\sum a(\vec{x}).p_a|\!) = \sum a(\vec{x}).(\!|p_a|\!). \\ \quad = x \mid \overline{a}\langle (\!|n_1|\!), \ldots, (\!|n_k|\!)\rangle & if \quad p \Downarrow x \mid \overline{a}\langle n_1, \ldots, n_k\rangle. & \end{array}$$

▶ **Proposition 11** (Associativity)**.** *Let d be a design and $n_1, \ldots, n_k$ be negative designs. One has:* $(\!|d[n_1/x_1, \ldots, n_k/x_k]|\!) = (\!|(\!|d|\!)[(\!|n_1|\!)/x_1, \ldots, (\!|n_k|\!)/x_k]|\!)$.

We finally define an orthogonality relation on so-called atomic designs.

▶ **Definition 12** (Atomic designs)**.** A positive standard design p is *atomic* if it has at most one free variable; that variable will be called x_0 in the rest of the paper. A negative standard design n is *atomic* if it is closed, *i.e.*, $\mathrm{fv}(n) = \emptyset$.

▶ **Definition 13.** Let p be a positive atomic design and n a negative atomic design. The designs p and n are said to be *orthogonal* (written $p \perp n$) if $p[n/x_0] \in \perp\!\!\!\perp$. Given a set $\mathbf{X}$ of atomic designs of the same polarity, we define its orthogonal $\mathbf{X}^\perp := \{\, e \mid \forall d \in \mathbf{X}, d \perp e \,\}$.

The orthogonality relation enjoys the usual properties:

▶ **Proposition 14.** *Let $\mathbf{X}$, $\mathbf{Y}$ be sets of atomic designs of the same polarity. One has:*

1) $\mathbf{X} \subseteq \mathbf{X}^{\perp\perp}$ 2) $\mathbf{X} \subseteq \mathbf{Y} \Rightarrow \mathbf{Y}^\perp \subseteq \mathbf{X}^\perp$ 3) $\mathbf{X}^\perp = \mathbf{X}^{\perp\perp\perp}$ 4) $(\mathbf{X} \cup \mathbf{Y})^\perp = \mathbf{X}^\perp \cap \mathbf{Y}^\perp$

3.3 Behaviours, Sets of Designs

Given a set $\mathbf{X}$ of atomic designs of the same polarity, $\mathbf{X}^\perp$ is the set of all those atomic designs that interact properly with respect to $\mathbf{X}$: they have a common behaviour with respect to the elements of $\mathbf{X}$. Proposition 14 ensures that such $\mathbf{X}^\perp$ are equal to their bi-orthogonal, this property characterizes them as *behaviours*:

▶ **Definition 15.** A *behaviour* is a set $\mathbf{X}$ of atomic designs of the same polarity such that $\mathbf{X} = \mathbf{X}^{\perp\perp}$. We denote by $\mathcal{C}_P$ (resp. $\mathcal{C}_N$) the set of all positive (resp. negative) behaviours.

$\mathcal{C}_P$, ordered by set inclusion, forms a complete lattice: using Proposition 14, we prove easily that every collection of positive behaviours $\vec{\mathbf{S}}$ has $(\bigcup \vec{\mathbf{S}})^{\perp\perp}$ as a least upper bound and $(\bigcap \vec{\mathbf{S}})^{\perp\perp} = \bigcap \vec{\mathbf{S}}$ as a greatest lower bound. Thus the Knaster-Tarski theorem guarantees the existence of least and greatest fixed points of monotonic operators on $\mathcal{C}_P$.

We generalize the relation $d \in \mathbf{C}$ between atomic designs and behaviours into the relation $d \models \mathbf{\Gamma}$ between designs with arbitrary free variables and contexts of behaviours:

▶ **Definition 16.** A *positive context* $\mathbf{\Gamma}$ is a set of pairs $x_1 : \mathbf{P_1}, \ldots, x_k : \mathbf{P_k}$ where $x_1, \ldots, x_k$ are distinct variables and $\mathbf{P_1}, \ldots, \mathbf{P_k}$ are positive behaviours. A *negative context* $\mathbf{\Gamma}, \mathbf{N}$ is a positive context $\mathbf{\Gamma}$ together with a negative behaviour $\mathbf{N}$, to which no variable is associated.

▶ **Definition 17.** Let $\mathbf{\Gamma} = x_1 : \mathbf{P_1}, \ldots, x_k : \mathbf{P_k}$ be a positive context, $\mathbf{\Gamma}, \mathbf{N}$ be a negative context, p (resp. n) be a positive (resp. negative) standard design. We define:

$$\begin{array}{llll} p \models \mathbf{\Gamma} & \text{iff} \quad p[n_1/x_1, \ldots, n_k/x_k] \in \perp\!\!\!\perp & \text{for any} & n_1 \in \mathbf{P_1}^\perp, \ldots, n_k \in \mathbf{P_k}^\perp; \\ n \models \mathbf{\Gamma}, \mathbf{N} & \text{iff} \quad p[n[n_1/x_1, \ldots, n_k/x_k]/x_0] \in \perp\!\!\!\perp & \text{for any} & p \in \mathbf{N}^\perp, n_1 \in \mathbf{P_1}^\perp, \ldots, n_k \in \mathbf{P_k}^\perp. \end{array}$$

Remark that $p \models x_0 : \mathbf{P}$ if and only if $p \in \mathbf{P}$ and $n \models \mathbf{N}$ if and only if $n \in \mathbf{N}$. More generally, the following closure principle [11, 19] will be useful in the following sections.

▶ **Proposition 18** (Closure principle).

$d \models \mathbf{\Sigma}, z : \mathbf{P}$	*iff*	$\forall m \in \mathbf{P}^{\perp},\ (\!|d[m/z]|\!) \models \mathbf{\Sigma}$	*where* $\mathbf{\Sigma}$ *is a positive or negative context.*
$n \models \mathbf{\Gamma}, \mathbf{N}$	*iff*	$\forall q \in \mathbf{N}^{\perp},\ (\!|q[n/x_0]|\!) \models \mathbf{\Gamma}$	*where* $\mathbf{\Gamma}$ *is a positive context.*

3.4 Designs as Strategies

We end this background section on ludics by providing some more details on the comparison between HO game semantics [12] and ludics.

In HO game semantics, one first defines arenas which specify the moves of the game and which induce plays. In a second step, strategies are defined, as sets of plays satisfying various conditions (such as totality, determinism, innocence, *etc.*) depending on what is modeled. While arenas interpret formulas (or types), strategies will interpret proofs (or programs).

In ludics, the construction proceeds in the other direction, more akin to realizability models: a notion of abstract proof (design) serves as our notion of strategy while arenas are replaced by behaviours, that are sets of designs closed under bi-orthogonality. Strategies and interaction thus come first and only afterwards come the notion of arena: the moves of the game are defined as a by-product of the way the objects interact.

The comparison can be made more precise when comparing innocent game semantics and ludics [9, 4]. Indeed, with *innocent* strategies, a player's move does not depend on the full play that precedes it but only on a restriction of the play, its *view*. A view typically excludes the part of the play which corresponds to intermediate computations of the opponent, retaining only opponent's results and not how the results were obtained. As a consequence, innocent strategies can be presented as *sets of views* with some conditions. Ludics fits this presentation as designs can be seen as sets of views: each branch of a design is a view.

To conclude this comparison, let us stress that on the one hand, game semantics puts constraints on the way arenas are built but it is then rather flexible on the definition of strategies (by enforcing or relaxing various constraints on the structure of strategies). On the other hand, ludics puts constraints on the design of strategies (for instance to preserve analytical theorems on which internal completeness depends) and is quite flexible on how arenas are defined. This difference explains why it revealed to be much more difficult to model LL exponentials in ludics than in HO game semantics. The very same reason explains why it will be smoother to interpret fixed points in ludics than in HO game semantics [7]. We will come back to this last point when discussing related works in Section 6.

4 Interpretation of μMALLP in Ludics

We now define a semantics for our system in ludics, extending the usual interpretation of MALL in computational ludics [19]: formulas will be interpreted by behaviours and proofs by designs. From now on, we restrict to the MALL signature from Definition 6.

4.1 Interpretation of Formulas

▶ **Definition 19.** Let F be a preformula and $\mathcal{E}$ an *environment* mapping each free variable of F to a behaviour of the same polarity. We define by induction on F a behaviour called the *interpretation of* F *under* $\mathcal{E}$ and denoted by $[\![F]\!]^{\mathcal{E}}$.

$[\![X]\!]^{\mathcal{E}} = \mathcal{E}(X) \quad [\![0]\!]^{\mathcal{E}} = \emptyset^{\perp\perp} \quad [\![1]\!]^{\mathcal{E}} = \{(x_0 \mid 1)\}^{\perp\perp} \quad [\![\downarrow N]\!]^{\mathcal{E}} = \{x_0 \mid \downarrow\langle r\rangle \,:\, r \in [\![N]\!]^{\mathcal{E}}\}^{\perp\perp}$
$[\![N_1{\otimes}N_2]\!]^{\mathcal{E}} = \{(x_0 \mid \otimes\langle r_1, r_2\rangle) \,:\, r_i \in [\![N_i]\!]^{\mathcal{E}}\}^{\perp\perp}$
$[\![N_1{\oplus}N_2]\!]^{\mathcal{E}} = \{(x_0 \mid \oplus_i\langle r_i\rangle) \,:\, i \in \{1,2\}, r_i \in [\![N_i]\!]^{\mathcal{E}}\}^{\perp\perp}$
$[\![\mu X.P]\!]^{\mathcal{E}} = \mathrm{lfp}(\Phi) \quad \text{where} \quad \Phi : \mathcal{C}_P \longrightarrow \mathcal{C}_P,\ \mathbf{C} \longmapsto [\![P]\!]^{\mathcal{E}, X \mapsto \mathbf{C}}$
$[\![N]\!]^{\mathcal{E}} = ([\![N^{\perp}]\!]^{\mathcal{E}})^{\perp} \quad \text{for all other cases}$

The interpretation of a formula F in the empty environment is simply written $[\![F]\!]$.

The well-definedness of the interpretation of $\mu X.P$ relies on the monotonicity of Φ, which is easily proved by induction on monotonic preformulas. Our interpretation of formulas enjoys the usual substitution property, which entails that the interpretation of fixed points is stable under unfolding.

▶ **Proposition 20.** $[\![F[G/X]]\!]^{\mathcal{E}} = [\![F]\!]^{\mathcal{E}, X \mapsto [\![G]\!]^{\mathcal{E}}}$ *and* $[\![\mu X.P]\!]^{\mathcal{E}} = [\![P[\mu X.P/X]]\!]^{\mathcal{E}}$.

The interpretation of positive MALL formulas is made by biorthogonal closure and that of negative MALL formulas is made by orthogonal closure, so that the shape of the elements of the resulting sets is not obvious. Nevertheless, the internal completeness theorem of ludics [19] allows to characterize them. For example, if $p = x_0 \mid \overline{a}\langle\vec{n}\rangle \in [\![N_1{\otimes}N_2]\!]$ then $\overline{a} = \otimes$, $\vec{n} = (n_1, n_2)$ and each $n_i \in [\![N_i]\!]$. In the same way, if $n = \sum a(\vec{x}_a).p_a \in [\![P_1 ⅋ P_2]\!]$ then $\vec{x}_{⅋} = (x_1, x_2)$ and $p_{⅋} \models x_1{:}[\![P_1]\!], x_2{:}[\![P_2]\!]$. The treatment of other MALL connectives can be found in Proposition 51 in the appendix. Similarly, the interpretation of a ν formula is defined as the orthogonal of a least fixed point, but that is equivalent to the following more direct description as a greatest fixed point.

▶ **Proposition 21.** $[\![\nu X.N]\!]^{\mathcal{E}} = \mathrm{gfp}(\Phi)$ *where* $\Phi : \mathcal{C}_N \to \mathcal{C}_N$ *is such that* $\Phi(\mathbf{C}) = [\![N]\!]^{\mathcal{E}, X \mapsto \mathbf{C}}$.

4.2 Interpretation of μMALLP Proofs

We interpret proofs compositionally, each rule corresponding to a construction on designs. Again, this extends the interpretation of MALL rules by Terui and Basaldella [5]. Proofs of positive sequents are going to be interpreted as positive designs, and similarly for negative sequents. In order to do so, we need to annotate positive formulas in sequents with distinct variable names. If $\Gamma = P_1, \ldots, P_n$ is a positive sequent, we say that $x_1 : P_1, \ldots, x_n : P_n$ is a *decoration* of Γ when the x_i are distinct positive variables. A decoration for a negative sequent Γ, N is obtained by adding N to a decoration of the positive part Γ.

We first give the final definition of the interpretation, in order to fix the ideas regarding its structure and purpose. Then we define the design construction $\mathbb{G}_{F,d}$ that is needed to interpret rule (ν).

▶ **Definition 22.** Let π be a proof of a sequent Γ, and Γ' a decoration of Γ. The interpretation of π in Γ' (written $[\![\pi]\!]^{\Gamma'}$) is defined by the rules of Figure 3. Each of these rules has the form

$$\frac{\{d_i \vdash \Gamma'_i\}_{i \in I}}{d \vdash \Gamma'}\ (r)$$

and stands for the following implication: If a proof π is obtained from the proofs $(\pi_i)_{i \in I}$ by applying rule (r), and $[\![\pi_i]\!]^{\Gamma'_i} = d_i$, then $[\![\pi]\!]^{\Gamma'} = d$.

This interpretation may be understood by thinking of designs as proof terms: formulas are annotated by variables and proofs by designs in the same way that, in intuitionistic logic, hypotheses are annotated by variables and proofs by λ-terms.

Identity rules

$$\dfrac{}{\eta_P[x/x_0] \vdash x : P, P^\perp}\ (ax) \qquad \dfrac{n \vdash \Gamma, P^\perp \qquad d \vdash \Delta, x : P}{(\!|d[n/x]|\!) \vdash \Gamma, \Delta}\ (cut)$$

Fixed point rules

$$\dfrac{p \vdash \Gamma, x : P[\mu X.P/X]}{p \vdash \Gamma, x : \mu X.P}\ (\mu) \qquad \dfrac{n \vdash x : S, N[S^\perp/X] \qquad m \vdash \Gamma, S^\perp}{(\!|\mathbb{G}_{N,n}[m/x_0]|\!) \vdash \Gamma, \nu X.N}\ (\nu)$$

MALL rules

$$\dfrac{}{\Omega^- \vdash \Gamma, \top}\ (\top) \qquad \dfrac{p_1 \vdash \Gamma, x_1 : P_1 \qquad p_2 \vdash \Gamma, x_2 : P_2}{\&_1(x_1).p_1 + \&_2(x_2).p_2 \vdash \Gamma, P_1 \& P_2}\ (\&) \qquad \dfrac{n \vdash \Gamma, N}{x \mid {\downarrow}\langle n\rangle \vdash \Gamma, x : {\downarrow}N}\ (\downarrow)$$

$$\dfrac{p \vdash \Gamma}{\bot.p \vdash \Gamma, \bot}\ (\bot) \qquad \dfrac{p \vdash \Gamma, x_1 : P_1, x_2 : P_2}{\parr(x_1, x_2).p \vdash \Gamma, P_1 \parr P_2}\ (\parr) \qquad \dfrac{p \vdash \Gamma, x : P}{{\uparrow}(x).p \vdash \Gamma, {\uparrow}P}\ (\uparrow)$$

$$\dfrac{}{(x \mid 1) \vdash x : 1}\ (1) \qquad \dfrac{n_1 \vdash \Delta, N_1 \qquad n_2 \vdash \Gamma, N_2}{x \mid \otimes\langle n_1, n_2\rangle \vdash \Gamma, \Delta, x : N_1 \otimes N_2}\ (\otimes) \qquad \dfrac{n \vdash \Gamma, N_i}{x \mid \oplus_i\langle n\rangle \vdash \Gamma, x : N_1 \oplus N_2}\ (\oplus_i)$$

Figure 3 Interpretation of μMALLP proofs.

The interpretation of rules (ax) and (cut) is quite natural. The axiom over P is interpreted by the copycat design η_P and cut is interpreted by the normal form of the cut between the interpretations of the two subproofs. The interpretation of MALL rules is the same as in [5]. The interpretation of the (μ) rule is trivial, based on the fact that fixed point unfolding is transparent in our interpretation. The main difficulty lies in the interpretation of the (ν) rule. Our goal is to interpret proofs by designs that reflect the computational behaviour of these proofs, thus we will derive the interpretation of rule (ν) from the cut reduction rule between μ and ν formulas presented in Figure 2. More precisely, our interpretation of rule (ν) is a design defined by an equality which expresses that the interpretation of the two proofs in Figure 2, which are obtained one from the other by cut elimination, are equal. As the reduction rule involves functoriality, our interpretation of rule (ν) is based on the construction $\mathbb{F}_{Q,d}$, the functoriality of Q applied to a design d, which is the counterpart in ludics of the construction $F_Q(\Pi)$, the functoriality of Q applied to a proof Π.

▶ **Definition 23.** Let d be a negative l-design and F a monotonic preformula such that $\mathrm{fv}(d) \subseteq \{x\}$ and $\mathrm{fv}(F) \subseteq \{X\}$. The *functoriality* of F applied to d is the negative l-design $\mathbb{F}_{F,d}$ coinductively defined by $\mathbb{F}_{F,d} = \eta_F$ when $fv(F) = \emptyset$, and otherwise:

$$\begin{array}{lllll}
\mathbb{F}_{X,d} & = & \mathbb{F}_{X^\perp,d} & = & d[x_0/x] \\
\mathbb{F}_{F_1 \otimes F_2,d} & = & \mathbb{F}_{F_1 \parr F_2,d} & = & \parr(x_1, x_2).(x_0 \mid \otimes\langle \mathbb{F}_{F_1,d}[x_1/x_0], \mathbb{F}_{F_2,d}[x_2/x_0]\rangle) \\
\mathbb{F}_{F_1 \oplus F_2,d} & = & \mathbb{F}_{F_1 \& F_2,d} & = & \sum_{i=1,2} \&_i(x_i).(x_0 \mid \oplus_i\langle \mathbb{F}_{F_i,d}[x_i/x_0]\rangle) \\
\mathbb{F}_{\downarrow F,d} & = & \mathbb{F}_{\uparrow F,d} & = & {\uparrow}(x_1).(x_0 \mid {\downarrow}\langle \mathbb{F}_{F,d}[x_1/x_0]\rangle) \\
\mathbb{F}_{\sigma Y.F,d} & = & \mathbb{F}_{F[\sigma Y.F/Y],d} & \text{for } \sigma \in \{\mu, \nu\} &
\end{array}$$

The definition of functoriality in ludics naturally expresses the intended computational behaviour of that operation: $\mathbb{F}_{Q,d}$ is a modified η-expansion which behaves as d on occurrences of X in Q. This should be contrasted with the very involved formulation of $F_Q(\Pi)$ in sequent calculus [2], which notably uses the (ν) rule to deal with fixed points encountered in Q.

▶ **Definition 24.** Let d be a negative design and F a monotonic preformula such that $\mathrm{fv}(d) \subseteq \{x\}$, $\mathrm{fv}(F) \subseteq \{X\}$ and $F \neq X$. The *action* of F on d is the design $\mathbb{G}_{F,d}$ coinductively defined by the following (productive) equation: $\mathbb{G}_{F,d} = \mathbb{F}_{F,\mathbb{G}_{F,d}}[d[x_0/x]/x_0]$.

4.3 Soundness and Invariance by Cut Elimination

Our first soundness result establishes that provability ($\vdash$) implies realizability ($\models$).

▶ **Definition 25.** If $\Gamma' = x_1 : P_1, \ldots, x_n : P_n$ is a positive decorated sequent, we interpret it into the positive context $[\![\Gamma']\!] = x_1 : [\![P_1]\!], \ldots, x_n : [\![P_n]\!]$. If Γ', N is a negative decorated sequent, its interpretation as a negative context is defined by $[\![\Gamma', N]\!] = [\![\Gamma']\!], [\![N]\!]$.

▶ **Theorem 26.** *If π is a proof of Γ, and Γ' is a decoration of Γ, then $[\![\pi]\!]^{\Gamma'} \models [\![\Gamma']\!]$.*

The theorem is proved by induction on π, and case analysis on its last rule. Soundness of rule (ax) follows from the fact that $\eta_P \models x_0 : [\![P]\!], [\![P]\!]^\perp$. Soundness for (cut) follows from the closure principle. The cases of MALL rules easily follow from the definition of formula interpretations. Soundness for rule (μ) is a direct consequence of Proposition 20. The difficulty lies in the (ν) rule, the soundness of which relies on the following proposition, proved in Appendix A.1.1, stating that $\mathbb{F}_{F,d}$ is sound.

▶ **Proposition 27.** *Let d be a negative design, $\mathbf{P}, \mathbf{N}$ be two behaviours and F be a negative monotonic preformula such that $\mathrm{fv}(d) \subseteq \{x\}$, $\mathrm{fv}(F) \subseteq \{X\} \subseteq \mathcal{V}_N$ and $d \models x : \mathbf{P}, \mathbf{N}$. Then we have $(\!|\mathbb{F}_{F,d}|\!) \models x_0 : [\![F^\perp]\!]^{X \mapsto \mathbf{P}^\perp}, [\![F]\!]^{X \mapsto \mathbf{N}}$.*

▶ **Proposition 28.** *Let $\nu X.F$ be a formula and d a design such that $d \models x_0 : \mathbf{S}, [\![F]\!]^{X \mapsto \mathbf{S}^\perp}$. We have $(\!|\mathbb{G}_{F,d}|\!) \models x_0 : \mathbf{S}, [\![\nu X.F]\!]$.*

Proof. By closure principle, one has that $(\!|\mathbb{G}_{F,d}|\!) \models x_0 : \mathbf{S}, [\![\nu X.F]\!]$ iff $\forall m \in \mathbf{S}^\perp, (\!|\mathbb{G}_{F,d}[m/x_0]|\!) \models [\![\nu X.F]\!]$ iff $\mathbf{S}_1 = \{\ (\!|\mathbb{G}_{F,d}[m/x_0]|\!)\ :\ m \in \mathbf{S}^\perp\ \}^{\perp\perp} \subseteq [\![\nu X.F]\!]$. But $[\![\nu X.F]\!] = \mathrm{gfp}(\phi)$ where $\phi = \mathbf{C} \mapsto [\![F]\!]^{X \mapsto \mathbf{C}}$, thus it suffices to establish that $\mathbf{S}_1$ is a post-fixed point of ϕ, ie $\mathbf{S}_1 \subseteq [\![F]\!]^{X \mapsto \mathbf{S}_1}$. This is equivalent to $\forall m \in \mathbf{S}^\perp, (\!|\mathbb{G}_{F,d}|\!)[m/x_0] \models [\![F]\!]^{X \mapsto \mathbf{S}_1}$ and by closure principle to $(\!|\mathbb{G}_{F,d}|\!) \models x_0 : \mathbf{S}, [\![F]\!]^{X \mapsto \mathbf{S}_1}$. Remark that by definition of $\mathbf{S}_1$ we have $(\!|\mathbb{G}_{F,d}|\!) \models x_0 : \mathbf{S}, \mathbf{S}_1$. By Proposition 27, this gives us $(\!|\mathbb{F}_{F,\mathbb{G}_{F,d}}|\!) \models x_0 : [\![F^\perp]\!]^{X \mapsto \mathbf{S}^\perp}, [\![F]\!]^{X \mapsto \mathbf{S}_1}$. By hypothesis, $d \models x_0 : \mathbf{S}, [\![F]\!]^{X \mapsto \mathbf{S}^\perp}$ so by the closure principle we have, as expected:

$$(\!|\mathbb{G}_{F,d}|\!) = (\!|\mathbb{F}_{F,\mathbb{G}_{F,d}}[d/x_0]|\!) \models x_0 : \mathbf{S}, [\![F]\!]^{X \mapsto \mathbf{S}_1}$$ ◀

As a second soundness result, we show that our semantics is denotational, *i.e.*, the interpretation is invariant by cut elimination. The proof of this theorem relies on the following lemma, proved in Appendix A.1.2, which expresses that ludics functoriality $\mathbb{F}_{F,d}$ is the semantical counterpart of the proof construction presented in Section 2.

▶ **Lemma 29.** *Let Π be a proof of $\vdash P, N$ and Q a negative monotonic preformula such that $\mathrm{fv}(Q) \subseteq \{X\} \subseteq \mathcal{V}_N$. One has $[\![F_Q(\Pi)]\!]^{x:Q^\perp[P^\perp/X],Q[N/X]} = (\!|\mathbb{F}_{Q,[\![\Pi]\!]^{x:P,N}}|\!)$.*

▶ **Theorem 30.** *If Π' is obtained from Π by μMALLP cut elimination rules, then $[\![\Pi]\!] = [\![\Pi']\!]$.*

5 On Completeness

Clearly, not all designs are interpretations of proofs, since some designs are not even recursive. More generally, it is highly non-trivial whether (or when) one can recover coinvariants from a design in order to finitely express it as a proof; indeed, coinvariants are completely hidden in the process of normalizing the interpretation of the (ν) rule. This is essentially the same difficulty that Girard encounters with second-order existential quantification in ludics [11], and which lead him to give a completeness result for Π_1 formulas only. In our setting, that would correspond to handle least fixed points only, which would be rather weak. Fortunately, we can do better thanks to our direct treatment of fixed points in the semantics.

We now introduce the class of essentially finite designs with respect to which we prove a completeness theorem in the rest of the paper.

▶ **Definition 31.** *Essentially finite designs* (EFD) are *inductively* defined by:

$$\begin{array}{lcl} p & ::= & (x \mid 1) \mid (x \mid \oplus_i\langle n\rangle) \mid (x \mid \otimes\langle n_1, n_2\rangle) \mid (x \mid \downarrow\langle n\rangle) \\ n & ::= & \bot.p_0 \mid \&_1(x_1).p_1 + \&_2(x_2).p_2 \mid \parr(x_1, x_2).p \mid \uparrow(x_1).p_1 \mid \eta_F \mid \Omega^- \end{array}$$

with $x_1 \in fv(p_1)$, $x_2 \in fv(p_2)$ and $x_1, x_2 \in fv(p)$.

Essentially finite designs perform a finite computation (this is the MALL part) followed by a copycat. Even though they are inductively defined, EFDs can be infinite. In pure MALLP, proofs correspond exactly to EFDs. But, despite the fact that μMALLP extends MALLP, it is not obvious that completeness holds for EFDs in μMALLP. This is because the interpretation of μMALLP formulas yields more complex behaviours than with MALLP. As we shall see, we can still obtain this theorem, but it requires a bit of work.

▶ **Theorem 32.** *Let d be an essentially finite design, let Γ be a sequent and Γ' be a decoration of Γ. We have: $d \models \mathbf{\Gamma'}$ iff $d = [\![\pi]\!]^{\Gamma'}$ for some μMALLP proof π of Γ.*

The proof of this theorem is by induction on the structure of the EFD, using internal completeness. The only problematic case is when the EFD is an η-expansion: one needs to prove that if $\eta_F \models x_0 : Q, P^\perp$ then $\eta_F = [\![\pi]\!]^{x_0:Q,P^\perp}$ where π is a proof of $\vdash Q, P^\perp$. Observe that if $\eta_F \models x_0 : Q, P^\perp$ then F, P and Q have the same infinitary unfolding, hence $\eta_F = \eta_P = \eta_Q$. Moreover, it is easy to prove that $\eta_P \models x_0 : Q, P^\perp$ iff $[\![P]\!] \subseteq [\![Q]\!]$. Using these two observations, the η-expansion case amounts to proving the following theorem:

▶ **Theorem 33** (Completeness for semantic inclusion)**.** *Let P, Q be two positive formulas such that $[\![P]\!] \subseteq [\![Q]\!]$. There is a proof π of $\vdash Q, P^\perp$ satisfying $[\![\pi]\!]^{x_0:Q,P^\perp} = \eta_P$.*

The remainder of this paper is dedicated to the proof of this result. In order to study the provability of semantic inclusions in μMALLP, we shall introduce an intermediate, infinitary proof system S_∞. We prove that it is sound and complete for semantic inclusions (*i.e.*, $[\![F]\!] \subseteq [\![G]\!]$ iff $F \vdash G$ is derivable in S_∞) and that we can translate any S_∞ proof to a μMALLP derivation whose ludics interpretation is a copycat. We shall establish by the way that derivability in S_∞ and semantic inclusion are decidable.

5.1 The Infinitary Proof System S_∞

Our system S_∞ deals with two-sided sequents that always feature exactly one formula on each side. We first introduce S_∞ pre-proofs that are only locally sound, and then equip them with a validity condition that ensures soundness. This construction, as well as the resulting system, are very close to Santocanale's circular proofs [18].

▶ **Definition 34.** S_∞ *pre-proofs* are trees coinductively generated from the rules of Figure 4. In that figure, $s(F_1, \dots, F_n)$ stands for a formula whose toplevel connective is a MALL connective s of arity n (*e.g.*, $\otimes$ has arity 2) and $\sigma \in \{\mu, \nu\}$. We say that a pre-proof is *fully justified* if it does not contain an application of rule (A).

$$\dfrac{}{F \vdash G}\,(A)$$

$$\dfrac{\{F_i \vdash G_i\}_{i\in[n]}}{s(F_1,\dots,F_n) \vdash\ s(G_1,\dots,G_n)}\,(s)$$

$$\dfrac{F[\sigma X.F/X] \vdash G}{\sigma X.F \vdash G}\,(\sigma_l)$$

$$\dfrac{F \vdash G[\sigma X.G/X]}{F \vdash \sigma X.G}\,(\sigma_r)$$

Figure 4 Infinitary proof system S_∞.

$$\dfrac{\dfrac{\dfrac{\dfrac{P(F\vdash G)}{F \vdash G}}{\uparrow F \vdash \uparrow G}\,(\uparrow) \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(F\vdash G)}{F \vdash G}}{\uparrow F \vdash \uparrow G}\,(\uparrow) \qquad \dfrac{P(\uparrow F \vdash H)}{\uparrow F \vdash H}\,(\uparrow)}{\uparrow F \otimes \uparrow F \vdash \uparrow G \otimes H}\,(\otimes)}{F \vdash \uparrow G \otimes H}\,(\mu_l)}{\uparrow F \vdash \uparrow(\uparrow G \otimes H)}\,(\uparrow)}{\uparrow F \vdash H}\,(\nu_r)}{}}{\uparrow F \otimes \uparrow F \vdash \uparrow G \otimes H}\,(\otimes)}{\dfrac{\uparrow F \otimes \uparrow F \vdash G}{F \vdash G}\,(\mu_l)}\,(\mu_r)$$

Figure 5 Example of an S_∞ proof.

The pre-proofs may be seen as η-expansions, but they are partial and unsound. Partiality comes from rule (A), which allows to make arbitrary assumptions. Unsoundness comes from the fact that, even without (A), pre-proofs are only locally sound. For example, $\mu X.\downarrow\uparrow X \vdash \downarrow \nu Y.\uparrow\downarrow Y$ admits a fully justified pre-proof.

▶ **Definition 35.** Let F, G be two formulas. $P(F \vdash G)$ is the S_∞ pre-proof of $F \vdash G$ coinductively defined by applying the first available rule in (σ_r), (σ_l), (s) or (A), and constructing the proofs of the premises $F_i \vdash G_i$ using the same construction $P(F_i \vdash G_i)$.

In other words, $P(F \vdash G)$ decomposes F and G as they agree on MALL connectives, giving priority to left unfolding of μ and ν. When MALL connectives become different, the pre-proof stops on an application of rule (A).

▶ **Example 36.** Let $F = \mu X.\uparrow X \otimes \uparrow X$, $G = \mu X.\uparrow X \otimes \nu Y.\uparrow(\uparrow X \otimes Y)$ and $H = \nu Y.\uparrow(\uparrow G \otimes Y)$. The pre-proof $P(F \vdash G)$ is given in Figure 5, where we have stopped expanding $P(F \vdash G)$ on sequents that have already occurred, explicitly showing the regular structure of the infinite proof. Note that this proof is fully justified.

We now turn to defining the validity condition that pre-proofs will have to satisfy in order to become proper proofs. Validity is based on parity conditions, as in Santocanale's work [18]. This requires a few preliminary definitions regarding subformulas. We denote by $\leq$ the subformula ordering, *i.e.*, $F \leq G$ if F is a subformula of G, and by $<$ the strict subformula ordering, *i.e.*, $F < G$ if $F \leq G$ and $F \neq G$.

▶ **Definition 37.** We define $\rightsquigarrow$ to be the least reflexive transitive relation on formulas such that: $s(F_1,\dots,F_n) \rightsquigarrow F_i$ and $\sigma X.F \rightsquigarrow F[(\sigma X.F)/X]$.

Note that, for a given F, there are only finitely many G such that $F \rightsquigarrow G$ (such formulas are in fact in bijection with the (open) subformulas of F). Also note that if $F \vdash G$ appears under $F' \vdash G'$ in a pre-proof, one has $F \rightsquigarrow F'$ and $G \rightsquigarrow G'$.

▶ **Proposition 38.** *For any cycle $F_1 \rightsquigarrow F_2 \rightsquigarrow \dots F_n \rightsquigarrow F_1$ there is some $i \in [1;n]$ such that $F_i \leq F_j$ for all $j \in [1;n]$.*

▶ **Definition 39** (Validity condition)**.** Let π be a pre-proof and γ an infinite branch of π. We define γ_r (resp. γ_l) to be the set of formulas appearing infinitely often on the right (resp. left) of sequents of γ. By Proposition 38, the elements of γ_r (resp. γ_l) have a minimum w.r.t. $\leq$; we note it $\min_r(\gamma)$ (resp. $\min_l(\gamma)$). It is easy to see that these minima are fixed point formulas. We say that the branch γ is *valid* if either $\min_l(\gamma)$ is a least fixed point or $\min_r(\gamma)$ is a greatest fixed point. We say that π is *valid* if all of its branches are valid.

▶ **Example 40.** The pre-proof in the previous example is valid, for the simple reason that on all branches, the formula F is unfolded infinitely often on the left of sequents. The pre-proof $P(\uparrow F \vdash H)$ would also be valid for the same reason, but $P(H \vdash \uparrow F)$ is not valid: the branch corresponding to taking the right of each tensor has only least fixed points on the right of its sequents and greatest fixed points on the left. Consider finally the pre-proof $P(G \vdash F)$. All of its branches that eventually always go to the right of tensors are invalid: on such branches, the minimum of formulas that occur infinitely often is H on the left of sequents and F on the right. All other branches, *i.e.*, those that go infinitely often to the left of tensors, are valid because G occurs infinitely often on the left of their sequents.

5.2 Completeness of S_∞

We first show that semantic inclusions are provable in S_∞: $[\![F]\!] \subseteq [\![G]\!]$ entails that $P(F \vdash G)$ is a valid, fully justified derivation. We prove that $P(F \vdash G)$ is fully justified by using internal completeness and Proposition 20. Proving its validity requires a few technical lemmas regarding the subformula ordering, that bridge the gap between the syntactic validity condition and the semantics of formulas.

▶ **Definition 41.** Let F, H be two preformulas, and X_0 a variable of the same polarity as H, not occurring in F nor H. We define $\mathcal{O}_H^{X_0}(F)$ as the unique preformula such that $\mathcal{O}_H^{X_0}(F)[H/X_0] = F$ and $H \not\leq \mathcal{O}_H^{X_0}(F)$. We shall simply write $\mathcal{O}_H(F)$ when the name of the variable is irrelevant or can be inferred from the context.

▶ **Proposition 42.** *Let F, H be two formulas such that $H < F$. For every MALL connective s and $\sigma \in \{\mu, \nu\}$, one has:*

- *If $F = s(F_1, \ldots, F_n)$ then $\mathcal{O}_H(s(F_1, \ldots, F_n)) = s(\mathcal{O}_H(F_1), \ldots, \mathcal{O}_H(F_n))$.*
- *If $F = \sigma Y.G$ then $\mathcal{O}_H(\sigma Y.G) = \sigma Y.\mathcal{O}_H(G)$ and unfolding F commutes with abstracting over H,* i.e., *$\mathcal{O}_H(G[(\sigma Y.G)/Y]) = \mathcal{O}_H(G)[\mathcal{O}_H(\sigma Y.G)/Y]$.*

The proof of Proposition 42 can be found in Appendix A.2.1.

▶ **Proposition 43.** *If $[\![F]\!] \subseteq [\![G]\!]$ then $P(F \vdash G)$ is a proof.*

Proof sketch, see appendix for details. We prove the contrapositive. If $P(F \vdash G)$ is not fully justified, it is easy to show that $[\![F]\!] \not\subseteq [\![G]\!]$. Assume now that $P(F \vdash G)$ is fully justified but not valid. Then our derivation has an infinite branch $\gamma = (\gamma_k)_{0 \leq k} = (F_k \vdash G_k)_{0 \leq k}$ such that $F_l = \min_l(\gamma) = \nu X_l.K_l$ and $F_r = \min_r(\gamma) = \mu X_r.K_r$. Let d be the design that acts as an η-expansion along γ, and ✠ elsewhere, and let d_i be its subdesign corresponding to the subbranch of γ rooted in γ_i. Let I be the indices such that γ_i is the conclusion of an unfolding of F_l. We can show that, for all $i \in I$, $d_i \in [\![F_l]\!]$. This is proved by showing that $\mathbf{A} = \{ d_i \mid i \in I \}^{\perp\perp}$ is a post-fixed point of $\phi : \mathbf{C} \mapsto [\![K_l]\!]^{X_l \mapsto \mathbf{C}}$, which amounts to proving that $\forall i \in I, d_i \in [\![K_l]\!]^{X_l \mapsto \mathbf{A}}$ or, equivalently, since $d_i = d_{i+1}$, that $\forall i \in I, d_{i+1} \in [\![\mathcal{O}_{F_l}(F_{i+1})]\!]^{X_l \mapsto \mathbf{A}}$. Generalizing this statement, we actually prove by induction that $\forall j \in \mathbb{N},\ d_j \in [\![\mathcal{O}_{F_l}(F_j)]\!]^{X_0 \mapsto \mathbf{A}}$, using Proposition 42. From there, we can show $d \in [\![F]\!]$ and, using a symmetry argument, $d \notin [\![G]\!]$, which concludes the proof. ◀

5.3 From S_∞ to μMALLP

We now prove that any valid fully-justified S_∞ proof can be transformed into a μMALLP proof. To prove this, we extend μMALLP with the rule (A) of Figure 4 and we call this system μMALLP*.

▶ **Definition 44.** Let π be a proof of a sequent s in S_∞ (resp. μMALLP*). We denote the set of sequents appearing in π as S_π, the conclusions of (A)-rules of π as $\mathcal{A}_\pi$ and call them the *assumptions* of π, and we let C_π be $S_\pi \setminus A_\pi$. The complexity of π is $\#\pi := card(C_\pi)$.

▶ **Definition 45.** Let F, G be two formulas and $\mathcal{H} \subseteq S_{P(F\vdash G)}$, $P(F \vdash G)^{\mathcal{H}}$ is the proof obtained from $P(F \vdash G)$ by replacing all the occurrences of the subtrees rooted in s by an assumption on s, for every $s \in \mathcal{H}$. (Notice that the subtrees rooted in s are all the same, and are equal to the tree $P(s)$.)

The result will follow from a slightly more general lemma:

▶ **Lemma 46.** *Let F, G be two formulas and $\mathcal{H} \subseteq S_{P(F\vdash G)}$. If $P(F \vdash G)^{\mathcal{H}}$ is valid then there is a proof π of $F \vdash G$ in μMALLP* such that $\mathcal{A}_\pi \subseteq \mathcal{H}$.*

Proof sketch (see Appendix A.2.2 for details). Let $s = F \vdash G$. The proof is by induction on $\#P(s)^{\mathcal{H}}$. Observe that if $\#P(s)^{\mathcal{H}} = 0$, then $s \in \mathcal{H}$ and the result obviously holds by using rule (A) on s in μMALLP*. In the inductive case, there are two possibilities:

1. There exist $s_1, s_2 \in C_{P(s)^{\mathcal{H}}}$ such that no occurrence of sequent s_1 appears above an occurrence of s_2 in $P(s)^{\mathcal{H}}$. In that case we decompose $P(s)^{\mathcal{H}}$ into $\Pi' = P(s)^{\mathcal{H}\cup\{s_2\}}$ and $\Pi'' = P(s_2)^{\mathcal{H}}$. Both Π' and Π'' have strictly smaller complexity than $P(s)^{\mathcal{H}}$. By induction hypothesis we obtain μMALLP* proofs π' of s and π'' of s_2, such that $\mathcal{A}_{\pi'} \subseteq \mathcal{H} \cup \{s_2\}$ and $\mathcal{A}_{\pi''} \subseteq \mathcal{H}$, which we plug together at the level of s_2 to get a μMALLP* proof of s.
2. Otherwise, we can find a valid branch containing all sequents appearing in $P(s)^{\mathcal{H}}$ (not as assumptions). This branch has either a least fixed point as minimum on the left of its sequents, or a greatest fixed point on the right. Assuming $\min_l(\gamma) = F_l = \mu X_l.K_l$ we decompose the proof at the unfoldings of F_l and design a suitable invariant in order to gather the pieces into a μMALLP* proof. ◀

When instanciating $\mathcal{H}$ to the empty set in Lemma 46 and remarking that a proof π in μMALLP* such that $\mathcal{A}_\pi = \emptyset$ is a μMALLP proof, we finally obtain:

▶ **Proposition 47.** *If $P(F \vdash G)$ is fully justified and valid then $F \vdash G$ is derivable in μMALLP.*

We can finally prove completeness for semantic inclusions.

Proof of Theorem 33. The result follows from Proposition 43 combined with a strengthening of Proposition 47 ensuring that $F \vdash G$ is provable by an η-expansion. To get this, notice that when we extend the syntax of designs, for every sequent s, by a negative constants A_s, and we interpret rule (A) by $\overline{A_s \vdash s}\ ^{(A)}$, the interpretation of the μMALLP* proof of $F \vdash G$ constructed in lemma 46 is a partial η-expansion: indeed, we show that it is a copycat design which mimics $P(F \vdash G)^{\mathcal{H}}$ until reaching a sequent from $\mathcal{H}$ where it plays a constant A_s for some s in $\mathcal{H}$ (See Proposition 52 in appendix for a detailed proof). As a corollary, when $\mathcal{H} = \emptyset$, this interpretation becomes a usual η-expansion, *i.e.*, without the constants A_s. ◀

5.4 Decidability of Semantic Inclusion

▶ **Proposition 48.** *If $P(F \vdash G)$ is fully justified and valid then $[\![F]\!] \subseteq [\![G]\!]$.*

Proof. We proceed essentially in the same way as in Lemma 46, proving the following generalization: Let F, G be two formulas and $\mathcal{H} \subseteq S_{P(F \vdash G)}$. If $P(F \vdash G)^{\mathcal{H}}$ is valid then, under the hypothesis that $\forall K \vdash L \in \mathcal{H}$, $[\![K]\!] \subseteq [\![L]\!]$, one has $[\![F]\!] \subseteq [\![G]\!]$. ◀

▶ **Proposition 49.** *Checking whether $P(F \vdash G)$ is valid is decidable.*

Decidability is proved by reducing the validity of all infinite branches to checking a finite number of combinations of elementary cycles, thanks to the fact that validity does not depend on the order in which elementary cycles are followed.

Proof. We extend the notations γ_r, γ_l to every finite path γ in $P(F \vdash G)$: γ_r (resp. γ_l) denotes the set of formulas appearing to the right (resp. left) of the sequents of γ. Then $\min(\gamma_r)$, $\min(\gamma_l)$ and the validity condition are the same as for infinite paths. Notice first that $P(F \vdash G)$ is not valid iff there exists a sequent s in $P(F \vdash G)$ and satisfying (P):

(P) There is an invalid finite path γ_1 in $P(s)$ from the root to an occurrence s.

Indeed, if there is such an s in $P(F \vdash G)$, and if γ_0 denotes a path in $P(F \vdash G)$ from the root to an occurrence of s, then the infinite path $\gamma_0\gamma_1^\omega$ is invalid. Conversely, let γ be an invalid infinite path in $P(F \vdash G)$, $F_l = \min(\gamma_l)$ and $G_r = \min(\gamma_r)$. There is a sequent s appearing infinitely often in γ such that the left-hand side of s is the formula F_l. As G_r appears infinitely often in γ_r, there is a finite sub-path of γ starting with s, ending with s and containing a sequent s' whose right-hand side is G_r. This finite path is obviously invalid, hence s satisfies (P).

We now prove that checking whether a sequent s satisfies (P) is decidable. Let $\delta_1, \dots, \delta_n$ be the paths from the root of $P(s)$ to an occurrence of s which are of the form $\delta_i = s\sigma_i s$ and $s \notin \sigma_i$. We observe that every path γ from the root of $P(s)$ to an occurrence of s is a concatenation of some δ_i where $i \in I \subseteq [n]$, hence checking (P) amounts to find some $I \subseteq [n]$ such that $\min_{i \in I}(\min_l(\delta_i))$ is a ν formula and $\min_{i \in I}(\min_r(\delta_i))$ is a μ formula. This is obviously decidable.

Finally, since the number of sequents appearing in $P(F \vdash G)$ is finite and (P) is decidable, we conclude that validity is a decidable property for $P(F \vdash G)$. ◀

▶ **Theorem 50.** *Let F, G be two formulas. Checking whether $[\![F]\!] \subseteq [\![G]\!]$ is decidable.*

6 Conclusion

Contributions. We have provided μMALLP with a denotational semantics in computational ludics. This construction is very natural, and did not require any change in the semantical framework to accommodate fixed points. Our interpretation gives an explicit formulation of the computational behaviour of μMALLP proofs as designs, which may provide a helpful alternative viewpoint to understand cut elimination in μMALLP. The fact that our model in ludics is relatively simple to work with has allowed us to venture into completeness investigations, a topic that is known to be tricky in presence of (co)induction. We have proved completeness for essentially finite designs using completeness of semantic inclusions for μMALLP . Technically, this last result uses, as an intermediate formalism, the infinitary system S_∞, which is very close to circular proofs with parity conditions. In order to prove completeness of μMALLP with respect to semantic inclusion, we proved completeness of μMALLP with respect to S_∞.

Related works. This last result is very much related to the work of Santocanale and Fortier [18, 10] who studied a circular proof system for a purely additive linear logic, equipped it with a cut elimination procedure, and gave a semantics of proofs in μ-bicomplete categories. Actually, the proof of Proposition 47 is inspired by Santocanale's argument [17] in his proof that circular proofs correspond to morphisms in μ-bicomplete categories. In fact, we could easily exploit his argument more generally, to translate to μMALLP a larger class of regular designs than just η-expansions.

Another obviously related work is Clairambault's game semantics for μLJ [7, 8], that is intuitionistic logic extended with least and greatest fixed points. In this semantics, he interprets (finite) proof objects as (infinite) winning strategies. More precisely, Clairambault first builds arenas with loops, simplifying McCusker's arenas for recursive types [15]. He then needs to equip the arenas with winning conditions for (finite and) infinite plays in such a way as to ensure that composition preserves totality. In ludics, the construction is simpler due to the fact that arenas, defined as behaviours, are rather secondary objects being generated from designs. Our construction is made particularly smooth by the fact that Terui's designs are general enough to interpret μMALL proofs and that the usual orthogonality (*i.e.*, interaction) of ludics is sufficient to forbid infinite chatterings that were causing the loss of totality in Clairambault's framework.

Facing the same difficulties as in our setting for getting completeness results for μLJ, Clairambault opts for a simpler approach in [8], proving a completeness result for $\mu\mathrm{LJ}^\omega$, an infinitary cut-free variant of μLJ. We can formulate and prove the same result in our framework. More generally, note that since Clairambault's game semantics for μLJ can be adapted to the linear case, it would be natural to compare precisely the interpretations of μMALL proofs in the two models.

Finally, our work is also related, though less closely, with the work of Brotherston and Simpson [6] who have recently explored the relationship between infinite, regular and finite proof systems for classical arithmetic, leaving open the question of the relative expressiveness of the regular and finite formalisms.

Future work. The most natural development of our work would be to extend Santocanale's work to the multiplicative case in order to obtain a circular presentation of full μMALLP. By doing so, we can hope to sharpen our completeness result on EFD by extended it to regular designs for which we conjecture a similar completeness theorem can be achieved. Another very interesting research direction would be to investigate under which conditions we can obtain a full abstraction result for our semantics.

Acknowledgments. We thank anonymous reviewers as well as Pierre Clairambault for their comments on this paper. The second author is funded by a grant from DIM RDM-IdF, Région Île–de-France. This work has been funded by ANR project ANR-14-CE25-0007 RAPIDO.

References

1 Andreas Abel, Brigitte Pientka, David Thibodeau, and Anton Setzer. Copatterns: programming infinite structures by observations. In *The 40th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL '13, Rome*, pages 27–38. ACM, 2013.

2 David Baelde. Least and greatest fixed points in linear logic. *ACM Transactions on Computational Logic*, 13(1), April 2012.

3 Gilles Barthe, Maria Joao Frade, Eduardo Giménez, Luis Pinto, and Tarmo Uustalu. Type-based termination of recursive definitions. *Mathematical Structures in Computer Science*, 14(01):97–141, 2004.

4 Michele Basaldella and Claudia Faggian. Ludics with repetitions (exponentials, interactive types and completeness). *Logical Methods in Computer Science*, 7(2), 2011.

5 Michele Basaldella and Kazushige Terui. On the meaning of logical completeness. In *Typed Lambda Calculi and Applications, 9th International Conference, TLCA 2009, Brasilia, Brazil*, pages 50–64, 2009.

6 James Brotherston and Alex Simpson. Sequent calculi for induction and infinite descent. *Journal of Logic and Computation*, 21(6):1177–1216, December 2011.

7 Pierre Clairambault. Least and greatest fixpoints in game semantics. In *12th International Conference on the Foundations of Software Science and Computational Structures (FOSSACS)*, volume 5504 of *Lecture Notes in Computer Science*, pages 16–31. Springer, March 2009.

8 Pierre Clairambault. Strong functors and interleaving fixpoints in game semantics. *RAIRO - Theor. Inf. and Applic.*, 47(1):25–68, 2013.

9 Claudia Faggian and Martin Hyland. Designs, disputes and strategies. In *Computer Science Logic, 16th International Workshop, CSL 2002, 11th Annual Conference of the EACSL, Edinburgh, Scotland, UK*, volume 2471 of *Lecture Notes in Computer Science*, pages 442–457. Springer, 2002.

10 Jérôme Fortier and Luigi Santocanale. Cuts for circular proofs: semantics and cut-elimination. In *Computer Science Logic 2013 (CSL 2013), Torino, Italy*, volume 23 of *LIPIcs*, pages 248–262. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2013.

11 Jean-Yves Girard. Locus solum. *Mathematical Structures in Computer Science*, 11:301–506, 2001.

12 Martin Hyland and Luke Ong. On full abstraction for PCF. *Information and Computation*, 163(2):285–408, December 2000.

13 Jean-Louis Krivine. Realizability in classical logic. *Panoramas et synthèses*, 27:197–229, 2009.

14 Ralph Matthes. Monotone (co)inductive types and positive fixed-point types. *ITA*, 33(4/5):309–328, 1999.

15 Guy McCusker. *Games and full abstraction for a functional metalanguage with recursive types*. CPHC/BCS distinguished dissertations. Springer, 1998.

16 N. P. Mendler. Inductive types and type constraints in the second order lambda calculus. *Annals of Pure and Applied Logic*, 51(1):159–172, 1991.

17 Luigi Santocanale. A calculus of circular proofs and its categorical semantics. Technical Report RS-01-15, BRICS, Department of Computer Science, University of Aarhus, May 2001.

18 Luigi Santocanale. A calculus of circular proofs and its categorical semantics. In *Foundations of Software Science and Computation Structures*, volume 2303 of *Lecture Notes in Computer Science*, pages 357–371. Springer, 2002.

19 Kazushige Terui. Computational ludics. *Theoretical Computer Science*, 412(20):2048–2071, 2011.

Contents

A Proofs

A.1 Interpretation of μMALLP in Ludics

▶ **Proposition 51** (Internal completeness). *Let N_1, N_2 be two negative formulas and $p = x_0 \mid \overline{a}\langle\vec{n}\rangle$ a positive l-design.*

$$p \in [\![N_1 \otimes N_2]\!] \text{ iff } \overline{a} = \otimes, \vec{n} = (n_1, n_2) \text{ and each } n_i \in [\![N_i]\!]$$
$$p \in [\![N_1 \oplus N_2]\!] \text{ iff } \overline{a} = \oplus_i, \vec{n} = n_i \text{ and } n_i \in [\![N_i]\!] \text{ for some } i$$
$$p \in [\![\downarrow N_1]\!] \text{ iff } \overline{a} = \downarrow, \vec{n} = n_1 \text{ and } n_1 \in [\![N_1]\!]$$
$$p \in [\![1]\!] \text{ iff } \overline{a} = 1, \vec{n} = \emptyset$$

Let P_1, P_2 be two positive formulas and $n = \sum a(\vec{x}_a).p_a$ a negative l-design.

$$n \in [\![P_1 \parr P_2]\!] \text{ iff } \vec{x}_\parr = (x_1, x_2) \text{ and } p_\parr \models x_1{:}[\![P_1]\!], x_2{:}[\![P_2]\!]$$
$$n \in [\![P_1 \& P_2]\!] \text{ iff } \vec{x}_{\&_i} = x_i \text{ and } p_{\&_i} \models x_i{:}[\![P_i]\!] \text{ for all } i \in \{1,2\}$$
$$n \in [\![\uparrow P_1]\!] \text{ iff } \vec{x}_\uparrow = x_1 \text{ and } p_\uparrow \models x_1 : [\![P_1]\!]$$
$$n \in [\![\bot]\!] \text{ always holds}$$

Note that these conditions constrain at most two designs in the sum; all others are arbitrary.

A.1.1 Soundness of the Interpretation

▶ **Proposition** (27). *Let d be a negative design, $\mathbf{P}, \mathbf{N}$ be two behaviours and F be a negative monotonic preformula such that $\mathrm{fv}(d) \subseteq \{x\}$, $\mathrm{fv}(F) \subseteq \{X\} \subseteq \mathcal{V}_N$ and $d \models x : \mathbf{P}, \mathbf{N}$. Then we have $(\!|\mathbb{F}_{F,d}|\!) \models x_0 : [\![F^\perp]\!]^{X \mapsto \mathbf{P}^\perp}, [\![F]\!]^{X \mapsto \mathbf{N}}$.*

Proof. We shall prove a generalized form of the proposition. Fix d, $\mathbf{P}$ and $\mathbf{N}$ as in the above statement.

We say that a list of preformulas $\vec{G}$ is *adequate* to a list of variables $\vec{X}$ if they have the same length and, for all i, G_i and X_i have the same polarity.

Let F be a negative monotonic preformula such that $\mathrm{fv}(F) \subseteq \vec{Y}$, and $\vec{U}$ be a list of monotonic preformulas that is adequate to $\vec{Y}$ and such that $\mathrm{fv}(\vec{U}) \subseteq \{X\}$. Finally, let $\vec{\mathbf{C}}$ and $\vec{\mathbf{E}}$ be lists of behaviours respectively adequate to $\vec{Y}$ and $\vec{Y}^\perp$ such that, for all $U_i \in \vec{U}$,

$$(\!|\mathbb{F}_{U_i,d}|\!) \models x_0 : \mathbf{C_i}, \mathbf{E_i} \quad \text{if } \mathbf{C_i} \text{ is positive}$$
$$(\!|\mathbb{F}_{U_i,d}|\!) \models x_0 : \mathbf{E_i}, \mathbf{C_i} \quad \text{if } \mathbf{C_i} \text{ is negative}$$

We shall establish that:

$$(\!|\mathbb{F}_{F[\vec{U}/\vec{Y}],d}|\!) \models x_0 : [\![F^\perp]\!]^{\vec{Y} \mapsto \vec{\mathbf{C}}^\perp}, [\![F]\!]^{\vec{Y} \mapsto \vec{\mathbf{E}}}$$

We proceed by induction on F. We show the only difficult case, which is when $F = \nu Z.G$. We set $V = F[\vec{U}/\vec{Y}]$, and set out to show that:

$$(\!|\mathbb{F}_{V,d}|\!) \models x_0 : [\![F^\perp]\!]^{\vec{Y} \mapsto \vec{\mathbf{C}}^\perp}, [\![F]\!]^{\vec{Y} \mapsto \vec{\mathbf{E}}}$$

Let us consider the behaviour $\mathbf{S}$ defined by:

$$\mathbf{S} := \{ \ (\!|(\mathbb{F}_{V,d})[m/x_0]|\!) \ : \ m \in ([\![F^\perp]\!]^{\vec{Y} \mapsto \vec{\mathbf{C}}^\perp})^\perp \ \}^{\perp\perp}$$

By the closure principle, it suffices to prove $\mathbf{S} \subseteq [\![F]\!]^{\vec{Y}\mapsto\vec{\mathbf{E}}}$. By Proposition 21, $[\![F]\!]^{\vec{Y}\mapsto\vec{\mathbf{E}}}$ is the greatest post-fixed point of $\phi := \mathbf{C} \mapsto [\![G]\!]^{\vec{Y}\mapsto\vec{\mathbf{E}},Z\mapsto\mathbf{C}}$. Thus, it suffices to show that $\mathbf{S}$ is a post-fixed point of ϕ, which finally amounts, by the closure principle, to prove that:

$$(\!|\mathbb{F}_{V,d}|\!) \models x_0 : [\![F^\perp]\!]^{\vec{Y}\mapsto\vec{\mathbf{C}}^\perp}, [\![G]\!]^{\vec{Y}\mapsto\vec{\mathbf{E}},Z\mapsto\mathbf{S}}$$

Or equivalently, by unfolding in the functoriality and interpretation:

$$(\!|\mathbb{F}_{G[\vec{U}/\vec{Y},V/Z],d}|\!) \models x_0 : [\![G^\perp]\!]^{\vec{Y}\mapsto\vec{\mathbf{C}}^\perp,Z\mapsto[\![F^\perp]\!]^{\vec{Y}\mapsto\vec{\mathbf{C}}^\perp}}, [\![G]\!]^{\vec{Y}\mapsto\vec{\mathbf{E}},Z\mapsto\mathbf{S}}$$

This is obtained by induction hypothesis on G, since by definition of $\mathbf{S}$ we have $(\!|\mathbb{F}_{V,d}|\!) \models x_0 : [\![F^\perp]\!]^{\vec{Y}\mapsto\vec{\mathbf{C}}^\perp}, \mathbf{S}$. ◀

A.1.2 Invariance by Cut Elimination

▶ **Lemma** (29). *Let Π be a proof of $\vdash P, N$ and Q a negative monotonic preformula such that $\mathrm{fv}(Q) \subseteq \{X\} \subseteq \mathcal{V}_N$. One has $[\![F_Q(\Pi)]\!]^{x:Q^\perp[P^\perp/X],Q[N/X]} = (\!|\mathbb{F}_{Q,[\![\Pi]\!]^{x:P,N}}|\!)$.*

Proof. We set $d = [\![\Pi]\!]^{x:P,N}$. The proof is by induction on the maximum depth of X in Q, following the structure of the definition of functoriality in [2]. The only difficult case is when $Q = \mu Y.R$. In this case, $F_Q(\Pi)$ is defined by the following derivation, where we use rule (ν) on $(\mu Y.R[N^\perp/X])^\perp$ with $(\mu Y.R)[P/X]$ as a coinvariant:

$$\dfrac{\dfrac{\dfrac{F_{R[(\mu Y.R)[P/X]/Y]}(\Pi)}{\vdash R[P/X][(\mu Y.R)[P/X]/Y], R^\perp[N^\perp/X][(\mu Y.R)[P/X]/Y]}}{\vdash (\mu Y.R)[P/X], R^\perp[N^\perp/X][(\mu Y.R)[P/X]/Y]}\,(\mu)}{\vdash (\mu Y.R)[P/X], (\mu Y.R)^\perp[N^\perp/X]}\,(\nu)$$

Then if we set $T = R[(\mu Y.R)[P/X]/Y]$ and $d' = [\![F_T(\Pi)]\!]^{x:T[P/X],T^\perp[N^\perp/X]}$, we have:

$$[\![F_Q(\Pi)]\!]^{x:Q[P/X],Q^\perp[N^\perp/X]} = (\!|\mathbb{G}_{R[Q/X],d'}|\!)$$

By induction hypothesis, $d' = (\!|\mathbb{F}_{R[\mu Y.R[P/X]],d}|\!)$. We need to prove that:

$$(\!|\mathbb{F}_{Q,d}|\!) = (\!|\mathbb{G}_{R[Q/X],d'}|\!)$$

The equality of these two designs follows from the existence of a "bisimulation" B containing those two designs. More precisely, a relation B on designs such that $((\!|\mathbb{F}_{Q,d}|\!), (\!|\mathbb{G}_{R[Q/X],\mathbb{F}_{T,d}}|\!)) \in B$ and if $(d_1, d_2) \in B$ then either:

1. $d_i = \sum a(\vec{x}).d_{ai}$ and $\forall a$, $(d_{a1}, d_{a2}) \in B$;
2. $d_i = x \mid \overline{a}\langle d_{i1}, \ldots, d_{in}\rangle$ and $\forall j$, $(d_{1j}, d_{2j}) \in B$.

It is not difficult to see that if such a relation exists then we get the equality.

We set $B = B_1 \cup B_2 \cup B_3 \cup B_4$ where:

- $B_1 = \{((\!|\mathbb{F}_{\mu Y.L,d}|\!), (\!|\mathbb{G}_{L^\perp[N^\perp/X],\mathbb{F}_{L[\mu Y L[P/X]/Y]},d}|\!))$:
  L is positive , $\mathrm{fv}(L) \subseteq \{X, Y\} \subseteq \mathcal{V}_P\}$;
- $B_2 = \{((\!|\mathbb{F}_{M[\mu Y.L/Y],d}|\!)$,
  $(\!|\mathbb{F}_{M^\perp[N^\perp/X],\mathbb{G}_{L^\perp[N^\perp/X],\mathbb{F}_{L[\mu Y L[P/X]/Y]},d}}[\mathbb{F}_{M[\mu Y.L[P/X]],d}/x_0])|\!))$:
  L, M are positive , $\mathrm{fv}(L), \mathrm{fv}(M) \subseteq \{X, Y\} \subseteq \mathcal{V}_P\}$;
- $B_3 = \{((\!|\mathbb{F}_{M[\mu Y.L/Y],d}|\!)$,
  $(\!|\mathbb{F}_{M[\mu Y.L[P/X]],d}[\mathbb{F}_{M^\perp[N^\perp/X],\mathbb{G}_{L^\perp[N^\perp/X],\mathbb{F}_{L[\mu Y L[P/X]/Y]},d}}/x_0])|\!))$:
  L is positive, M is negative, $\mathrm{fv}(L), \mathrm{fv}(M) \subseteq \{X, Y\} \subseteq \mathcal{V}_P\}$;

- $B_4 = \{(d, d) : d$ is a standard design $\}$.

  The relation B verifies the conditions above. ◀

▶ **Theorem** (30). *If Π' is obtained from Π by the cut elimination rules given in [2], then $[\![\Pi]\!] = [\![\Pi']\!]$.*

Proof. The auxiliary cases together with the main cases for MALL are easy to treat and follow essentially from the associativity of design normalization. The only technical case is the one of Figure 2. Let us show how to treat one such reduction $\Pi_1 \to \Pi_2$. Let Γ' and Δ' be two annotated sequents for Γ and Δ respectively.

We set $d := [\![\Pi_L]\!]^{x:P[\mu X.P/X],\Gamma'}$, $n := [\![\Theta]\!]^{y:S^\perp,P^\perp[S^\perp/X]}$ and $m := [\![\Pi_R]\!]^{\Delta',S}$. Notice first that $\mathbb{G}_{P^\perp,n} = \mathbb{G}_{P,n}$. We have $[\![\Pi_1]\!]^{\Gamma',\Delta'} = (\!|d[(\!|\mathbb{G}_{P,n}[m/x_0]|\!)/x]|\!)$ and by Lemma 29 $[\![\Pi_2]\!]^{\Gamma',\Delta'} = (\!|d[\mathbb{F}_{P,(\!|\mathbb{G}_{P,n}|\!)}[n[m/y]/x_0]/x]|\!)$.

But by definition of $\mathbb{G}_{P,n}$ we have $\mathbb{G}_{P,n}[y/x_0] = \mathbb{F}_{P,\mathbb{G}_{P,n}}[n/x_0]$, thus $(\!|\mathbb{G}_{P,n}[y/x_0]|\!) = (\!|\mathbb{F}_{P,(\!|\mathbb{G}_{P,n}|\!)}[n/x_0]|\!)$. The equality of the two interpretations follows from this remark and from associativity of designs normalization. ◀

A.2 Completeness of μMALLP for EFD

A.2.1 Completeness of S_∞

▶ **Proposition** (42). *Let F, H be two formulas such that $H < F$. For every MALL connective s and $\sigma \in \{\mu, \nu\}$, one has:*

- *If $F = s(F_1, \dots, F_n)$ then $\mathcal{O}_H(s(F_1, \dots, F_n)) = s(\mathcal{O}_H(F_1), \dots, \mathcal{O}_H(F_n))$.*
- *If $F = \sigma Y.G$ then $\mathcal{O}_H(\sigma Y.G) = \sigma Y.\mathcal{O}_H(G)$ and unfolding F commutes with abstracting over H,* i.e., *$\mathcal{O}_H(G[(\sigma Y.G)/Y]) = \mathcal{O}_H(G)[\mathcal{O}_H(\sigma Y.G)/Y]$.*

Proof.

- If $F = s(F_1, \dots, F_n)$ then $E := s(\mathcal{O}_H(F_1), \dots, \mathcal{O}_H(F_n))$ verifies that $E[H/X_0] = F$. $H \not\leq E$, otherwise we would have either $H = E$ and then

$$H = H[H/X_0] = s(\mathcal{O}_H(F_1)[H/X_0], \dots, \mathcal{O}_H(F_n)[H/X_0]) = F$$

  which is not possible; or $H \leq \mathcal{O}_H(F_i)$ for some i which is not possible by definition.
- If $F = \sigma Y.G$, we apply the same argument to $E := \sigma Y.\mathcal{O}_H(G)$.
- We check that the right-hand side $E := \mathcal{O}_H(G)[\mathcal{O}_H(\sigma Y.G)/Y]$ verifies the two conditions of Definition 41. The first one is obvious: $E[H/X_0] = G[\sigma Y.G/Y]$. It remains to check that $H \not\leq E$. Since we obviously have $H \not\leq \mathcal{O}_H(G)$ and $H \not\leq \mathcal{O}_H(\sigma Y.G)$, it only remains to consider the case where $\mathcal{O}_H(\sigma Y.G) \leq H$. But, since X_0 does not occur free in H, this would mean that X_0 is not free in $\mathcal{O}_H(\sigma Y.G)$, which is equivalent to $H \not\leq \sigma Y.G$, contradicting our hypothesis. ◀

▶ **Proposition** (43). *If $[\![F]\!] \subseteq [\![G]\!]$ then $P(F \vdash G)$ is a proof.*

Proof. We show the contrapositive. If $P(F \vdash G)$ is not fully justified, it is easy to show that $[\![F]\!] \not\subseteq [\![G]\!]$. Assume now that $P(F \vdash G)$ is fully justified but not valid. We shall build a design d_0 such that (**P1**) $d_0 \in [\![F]\!]$ and (**P2**) $d_0 \notin [\![G]\!]$.

Since $P(F \vdash G)$ is not valid, it has an infinite branch $\gamma = (\gamma_k)_{0 \leq k}$ such that $F_l = \min_l(\gamma) = \nu X_l.K_l$ and $F_r = ob \min_r(\gamma) = \mu X_r.K_r$.

We set $\gamma_k = F_k \vdash G_k$ and we associate to every γ_k a design d_k coinductively defined by the following equations:

- If $F_k = \sigma X.E$ or $G_k = \sigma X.H$ where $\sigma \in \{\mu, \nu\}$, then $d_k = d_{k+1}$.

- If $F_k = F_{k_1} \parr F_{k_2}$, $G_k = G_{k_1} \parr G_{k_2}$, $F_{k+1} = F_{k_i}$, $G_{k+1} = G_{k_i}$ then $d_k = \parr(x_1, x_2).d_{k+1}[x_i/x_0]$.
- If $F_k = F_{k_0} \oplus F_{k_1}$ and $G_k = G_{k_0} \oplus G_{k_1}$, $F_{k+1} = F_{k_i}$, $G_{k+1} = G_{k_i}$ then $d_k = x_0 \mid \oplus_i \langle d_{k+1} \rangle$.
- We proceed similarly for $\otimes$, $\&$ and shifts.

For every k, the design d_k "follows" the branch starting from γ_k, in particular, d_0 can be seen as a design representing the branch γ.

(**P1**) We prove in the following that $d_0 \in [\![F]\!]$. Let $I = \{ m \; : \; F_m = F_l, \; F_{m+1} = K_l[F_l/X_l] \}$. First remark that by applying internal completeness iteratively, we have that for all k, $d_0 \in [\![F_0]\!] \Leftrightarrow d_1 \in [\![F_1]\!] \Leftrightarrow \cdots \Leftrightarrow d_k \in [\![F_k]\!]$. In particular for some $i \in I$, one has $d_0 \in [\![F]\!] \Leftrightarrow d_i \in [\![F_i]\!] = [\![F_l]\!]$. But $[\![F_l]\!] = \mathrm{gfp}(\phi)$ where $\phi : \mathbf{C} \mapsto [\![K_l]\!]^{X_l \mapsto \mathbf{C}}$, hence, to prove that $d_i \in [\![F_l]\!]$ it suffices to find a post-fixed point $\mathbf{A}$ of ϕ such that $d_i \in \mathbf{A}$. We shall establish this for $\mathbf{A} = \{ d_i \mid i \in I \}^{\perp\perp}$.

Showing that $\mathbf{A}$ is a post-fixed point amounts to showing that for every $m \in I$, $d_{m+1} \in [\![K_l]\!]^{X_l \mapsto \mathbf{A}}$, or equivalently $d_{m+1} \in [\![\mathcal{O}_{F_l}(F_{m+1})]\!]^{X_0 \mapsto \mathbf{A}}$ since $\mathcal{O}_{F_l}(F_{m+1}) = K_l[X_0/X_l]$.

More generally, we shall prove that $\forall m, n \in I$, $\forall i \in [m, n]$, $d_i \in [\![\mathcal{O}_{F_l}(F_i)]\!]^{X_0 \mapsto \mathbf{A}}$. We proceed by a decreasing induction on i. If $F_i = F_l$ (*i.e.*, when $i \in I$, and in particular when $i = n$) we have $\mathcal{O}_{F_l}(F_i) = X_0$, from which $d_i \in [\![X_0]\!]^{X_0 \mapsto \mathbf{A}}$ immediately follows. Otherwise, we distinguish several cases:

1. If $F_i = F_{i+1}$ then $d_i = d_{i+1}$. By induction hypothesis, $d_{i+1} \in [\![\mathcal{O}_{F_l}(F_{i+1})]\!]^{X_0 \mapsto \mathbf{A}}$ and hence $d_i \in [\![\mathcal{O}_{F_l}(F_i)]\!]^{X_0 \mapsto \mathbf{A}}$.
2. If $F_i = \sigma Y.E$ and $F_{i+1} = E[F_i/Y]$, we have $d_i = d_{i+1}$. It suffices to prove that $[\![\mathcal{O}_{F_l}(F_{i+1})]\!]^{X_0 \mapsto \mathbf{A}} = [\![\mathcal{O}_{F_l}(F_i)]\!]^{X_0 \mapsto \mathbf{A}}$. Since $F_l < F_i$, Proposition 42 yields $\mathcal{O}_{F_l}(F_i) = \sigma Y.\mathcal{O}_{F_l}(E)$ and:

$$\mathcal{O}_{F_l}(E[F_i/Y]) = \mathcal{O}_{F_l}(E)[\mathcal{O}_{F_l}(F_i)/Y]$$

   This allows us to conclude:

$$\begin{aligned} [\![\mathcal{O}_{F_l}(F_{i+1})]\!]^{X_0 \mapsto \mathbf{A}} &= [\![\mathcal{O}_{F_l}(E)[\mathcal{O}_{F_l}(F_i)/Y]]\!]^{X_0 \mapsto \mathbf{A}} \\ &= [\![\mathcal{O}_{F_l}(E)[\sigma Y.\mathcal{O}_{F_l}(E)/Y]]\!]^{X_0 \mapsto \mathbf{A}} \\ &= [\![\sigma Y.\mathcal{O}_{F_l}(E)]\!]^{X_0 \mapsto \mathbf{A}} \\ &= [\![\mathcal{O}_{F_l}(F_i)]\!]^{X_0 \mapsto \mathbf{A}} \end{aligned}$$

3. Otherwise F_i starts with a MALL connective. All connectives are treated similarly, and we only show the case where $F_i = F_{i+1} \parr E$. In that case, we recall that $d_i = \parr(x_0, x_1).d_{i+1}$. By induction hypothesis one has $d_{i+1} \in [\![\mathcal{O}_{F_l}(F_{i+1})]\!]^{X_0 \mapsto \mathbf{A}}$. Therefore, for every preformula H such that $\mathrm{fv}(H) \subseteq \{X_0\}$ one has:

$$\parr(x_0, x_1).d_{i+1} \in [\![\mathcal{O}_{F_l}(F_{i+1}) \parr H]\!]^{X_0 \mapsto \mathbf{A}}$$

   In particular we set $H = \mathcal{O}_{F_l}(E)$. By Proposition 42 we have $\mathcal{O}_{F_l}(F_i) = \mathcal{O}_{F_l}(F_{i+1}) \parr \mathcal{O}_{F_l}(E)$ which allows us to conclude: $d_i \in [\![\mathcal{O}_{F_l}(F_i)]\!]^{X_0 \mapsto \mathbf{A}}$.

(**P2**) We show now that $d_0 \notin [\![G]\!]$. To do so, we construct a design d'_0 such that $d'_0 \in [\![G]\!]^\perp$ and $d_0 \not\perp d'_0$. Let us consider the branch γ' of $P(G^\perp \vdash F^\perp)$ that follows the same occurrences as γ. We have $\min_l(\gamma') = (\min_r(\gamma))^\perp$, so it is a greatest fixed point. We can thus construct as before a design d'_0 such that $d'_0 \in [\![G]\!]^\perp$. It is easy to verify that the interaction between d_0 and d'_0 diverges and therefore that $d_0 \not\perp d'_0$. ◀

A.2.2 From S_∞ to μMALLP

▶ **Lemma** (46). *Let F, G be two formulas and $\mathcal{H} \subseteq S_{P(F\vdash G)}$. If $P(F \vdash G)^{\mathcal{H}}$ is valid then there is a proof π of $F \vdash G$ in μMALLP* such that $\mathcal{A}_\pi \subseteq \mathcal{H}$.*

Proof. Let $s = F \vdash G$. The proof is by induction on $\#P(s)^{\mathcal{H}}$. Observe that if $\#P(s)^{\mathcal{H}} = 0$, then $s \in \mathcal{H}$ and the result obviously holds by using rule (A) on s in μMALLP*. Otherwise, we distinguish two cases.

Case 1: The proof is not "strongly connected". There exist $s_1, s_2 \in C_{P(s)^{\mathcal{H}}}$ such that no occurrence of sequent s_1 appears above an occurrence of s_2 in $P(s)^{\mathcal{H}}$. In that case we decompose $P(s)^{\mathcal{H}}$ into $\Pi' = P(s)^{\mathcal{H}\cup\{s_2\}}$ and $\Pi'' = P(s_2)^{\mathcal{H}}$. Both Π' and Π'' have strictly smaller complexity than $P(s)^{\mathcal{H}}$ (because $s_2 \notin C_{\Pi'}$ and $s_1 \notin C_{\Pi''}$). By induction hypothesis we obtain μMALLP* proofs π' of s and π'' of s_2, such that $\mathcal{A}_{\pi'} \subseteq \mathcal{H} \cup \{s_2\}$ and $\mathcal{A}_{\pi''} \subseteq \mathcal{H}$. Let π be the proof obtained from π' by replacing all occurrences of rule (A) of conclusion s_2 by π''. Then π is a μMALLP* proof of s with $\mathcal{A}_\pi \subseteq \mathcal{H}$, from which we conclude.

Case 2: The proof is "strongly connected". Otherwise, given any two sequents $s_1, s_2 \in C_{P(s)^{\mathcal{H}}}$, for any occurrence of a sequent s_1 there is an occurrence of s_2 which is above s_1 in $P(s)^{\mathcal{H}}$. In this case, it is not possible to apply the same technique as above.

Instead we notice that the condition allows us to construct an infinite branch γ of $P(s)^{\mathcal{H}}$ containing all the elements of $C_{P(s)^{\mathcal{H}}}$ infinitely often. Since $P(s)^{\mathcal{H}}$ is valid, γ has either a least fixed point as minimum on the left of its sequents, or a greatest fixed point on the right. Let us treat the case where $\min_l(\gamma) = F_l = \mu X_l.K_l$, the other one being symmetric. We will choose to decompose $P(s)^{\mathcal{H}}$ at sequents where F_l was unfolded, that is, let $\mathcal{H}_l$ be the set of sequents which are conclusions of rules of the form:

$$\dfrac{K_l[F_l/X_l] \vdash K}{F_l \vdash K}\ (\mu_l)$$

We decompose $P(s)^{\mathcal{H}}$ at those sequents, considering the trees $\Pi' := P(s)^{\mathcal{H}\cup\mathcal{H}_l}$ and the trees $\Pi_K := P(K_l[F_l/X_l] \vdash K)^{\mathcal{H}\cup\mathcal{H}_l}$ for every $K \in \mathcal{R}_l = \{K \text{ such that } F_l \vdash K \in \mathcal{H}_l\}$.

Notice that $\#\Pi' < \#\Pi$ and $\#\Pi_K < \#\Pi$ regardless of K, so that we can apply the induction hypothesis and get θ' and θ_K respectively.

While it would be tempting to glue θ' and θ_K together, that would result in an infinite tree, that is not in a μMALLP proof. Instead, we will use the fact that F_l is a least fixed point and pick a suitable invariant, S, for the induction on F_l:

$$S = F_l \& (\mathop{\&}_{K\in\mathcal{R}_l} K)$$

this invariant is the with of all the the formulas appearing in the right of a sequent of $\mathcal{H}_l$ (and of F_l itself in order to keep the invariant $\mathcal{A}_\pi \subseteq \mathcal{H}$).

Thus, we replace in θ' every sequent $F_l \vdash K$ of $\mathcal{H}_l$ by the following derivation:

$$\dfrac{\dfrac{\{\ K_l[S/X_l] \vdash L\ \}_{L\in\mathcal{H}_l} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{F_l \vdash F_l}\ (ax)}{S \vdash F_l}\ (\&)}{K_l[S/X_l] \vdash K_l[F_l/X_l]}\ (K_l)}{K_l[S/X_l] \vdash F_l}\ (\mu_r)}{K_l[S/X_l] \vdash S}\ (\&) \qquad \dfrac{\dfrac{}{K \vdash K}\ (ax)}{S \vdash K}\ (\&)}{F_l \vdash K}\ (\mu)$$

which contains an application of (K_l), the functoriality on K_l. In order to get a μMALLP proof π with $\mathcal{A}_\pi \subseteq \mathcal{H}$, we must justify the sequents $K_l[S/X_l] \vdash L$ with proofs having all their assumptions in $\mathcal{H}$. We cannot directly use θ_L since its conclusion is $K_l[F_l/X_l] \vdash L$. Instead, we consider Π'_L which is obtained by replacing in Π_L every occurrence of F_l by the invariant S. (Remark that Π'_L contains assumptions of the form $S \vdash K$ instead of $F_l \vdash K$.) Π'_L has the same complexity as Π_L and induction hypothesis gives a proof θ'_L (having $S \vdash K$ among its assumptions).

The assumptions of conclusion $S \vdash K$ can be replaced by:

$$\dfrac{\dfrac{}{K \vdash K}\,(ax)}{S \vdash K}\,(\&)$$

from which we obtain a proof π_L that we use to justify the assumptions $K_l[S/X_l] \vdash L$ of θ getting derivation π.

There is a last step of transformation which is the reason why we added F_l to the invariant. Indeed, in the proof Π'_L, the elements of $\mathcal{H}_l$ are not the only ones affected by the substitution of F_l by S in Π_L: there are also the elements of $\mathcal{H}$. Indeed, this substitution has the effect of transforming each element $I \vdash J$ of $\mathcal{H}$ into $\mathcal{O}_{F_l}(I)[S/X_0] \vdash J$. Hence the assumptions of θ'_L (and, consequently, those of π_L) do not belong to $\mathcal{H}$, which is an invariant we want to keep. To force the assumptions of π_L to be in $\mathcal{H}$, we replace each assumption $\mathcal{O}_{F_l}(I)[S/X_0] \vdash J$ by:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{F_l \vdash F_l}\,(ax)}{S \vdash F_l}\,(\&)}{\mathcal{O}_{F_l}(I)[S/X_0] \vdash \mathcal{O}_{F_l}(I)[F_l/X_0]}\,(\mathcal{O}_{F_l}(I)) \qquad \dfrac{}{I = \mathcal{O}_{F_l}(I)[F_l/X_0] \vdash J}\,(A)}{\mathcal{O}_{F_l}(I)[S/X_0] \vdash J}\,(cut)$$

The resulting proof has its assumptions in $\mathcal{H}$, as expected. ◀

▶ **Proposition 52.** *Let F and G be two formulas such that $[\![F]\!] \subseteq [\![G]\!]$. The interpretation of the proof of $\vdash F^\perp, G$ constructed in Lemma 46 is η_F.*

Proof sketch. We add to the syntax of designs, for every sequent s a negative constant A_s and we interpret rule (A) as follows:

$$\dfrac{}{A_{\vdash F^\perp, G} \vdash x : F^\perp, G}\,(A)$$

Actually, we can see that the proofs of μMALLP* as proof contexts. Indeed, we can consider an application of an assumption rule as a hole. If s is a sequent used as an assumption inside a μMALLP* proof π, and if θ is another μMALLP* proof, we denote by $\pi[\theta/s]$ the proof obtained by replacing the occurrences of s by θ in π. We prove easily by induction on π that $[\![\pi[\theta/s]]\!] = [\![\pi]\!][[\![\theta]\!]/A_s]$, using associativity in the case of cut.

Next, we define the interpretation of S_∞ proofs in the natural way: rules (s) are interpreted by η-expansion steps, unfolding rules are transparent for the interpretation, and assumptions are interpreted by our new assumption constants.

For each proof $P(F \vdash G)^{\mathcal{H}}$, let $\pi(F \vdash G)^{\mathcal{H}}$ be the μMALLP* proof obtained from it in the proof of Lemma 46.

We prove that $[\![P(F \vdash G)^{\mathcal{H}}]\!] = [\![\pi(F \vdash G)^{\mathcal{H}}]\!]$ by the same induction as in the proof of Lemma 46; we use in the following the same notations as in this proof.

In Case 1 we have that $\Pi = \Pi'[\Pi''/s_2]$ and $\pi = \pi'[\pi''/s_2]$, thus $[\![\Pi]\!] = [\![\Pi']\!][[\![\Pi'']\!]/A_{s_2}]$ and $[\![\pi]\!] = [\![\pi']\!][[\![\pi'']\!]/A_{s_2}]$, but by induction hypothesis $[\![\Pi']\!] = [\![\pi']\!]$ and $[\![\Pi'']\!] = [\![\pi'']\!]$ hence $[\![\Pi]\!] = [\![\pi]\!]$.

Case 2 is more involved. In order to simplify the presentation, we shall assume that the invariant contains only two formulas, that is $\mathcal{R}_l = \{K_1, K_2\}$. Further we do not consider the inclusion of $F_l = \mu X_l.K_l$ into S. Thus $S = \downarrow(K_1 \& K_2)$ (we have to add a shift to adjust polarities). Finally, we directly consider the case where conclusion of the S_∞ derivation Π is $F_l \vdash K_1$. We claim that these are not essential restrictions. To prove the result, we define a bisimulation R between designs containing $(\llbracket\pi\rrbracket, \llbracket\Pi\rrbracket)$, in the same way as in the proof of the invariance under cut elimination. Since π starts with a ν rule, we have $\llbracket\pi\rrbracket = \mathbb{G}_{K_l,n}[m/x_0]$, where n is the interpretation of the subderivation establishing $\vdash K_l^\perp[S^\perp/X_l], x_0 : S$, and m is the interpretation of the subderivation establishing $\vdash S^\perp, x_0 : K_1$. If we denote $d := \mathbb{G}_{K_l,n}$ and $d_A := \mathbb{F}_{A,d}$ for any preformula A (so that $d = d_{K_l}[n/x_0]$) we pose $R := R_1 \cup R_2 \cup R_3$ where:

- $(\!|\llbracket\theta'_s\rrbracket[d_P/x]|\!)\ R_1\ \llbracket P(s)^{\mathcal{H}'}\rrbracket[x_0/x]$ if $s = (\vdash P[S^\perp/X_l], N)$ appears in $P(\vdash K_l^\perp[S^\perp/X_l], K_i)^{\mathcal{H}'}$ and $fv(\llbracket\theta'_s\rrbracket) = \{x\}$;
- $(\!|d_N[\llbracket\theta'_s\rrbracket/x_0]|\!)\ R_2\ \llbracket P(s)^{\mathcal{H}'}\rrbracket$ if $s = (\vdash N[S^\perp/X_l], P)$ appears in $P(\vdash K_l^\perp[S^\perp/X_l], K_i)^{\mathcal{H}'}$;
- $\llbracket\pi\rrbracket\ R_3\ \llbracket\Pi\rrbracket$.

This relation is a bisimulation. Checking the third item relies on the first one. The first one relies on the second one. Finally, the second one relies on the first, or the third when $N = X_l^\perp$. ◀