



HAL
open science

Concurrent Realizability on Conjunctive Structures

Emmanuel Beffara, Félix Castro, Mauricio Guillermo, Étienne Miquey

► **To cite this version:**

Emmanuel Beffara, Félix Castro, Mauricio Guillermo, Étienne Miquey. Concurrent Realizability on Conjunctive Structures. FSCD 2023 - 8th International Conference on Formal Structures for Computation and Deduction, Jul 2023, Rome, Italy. hal-04083002v1

HAL Id: hal-04083002

<https://inria.hal.science/hal-04083002v1>

Submitted on 27 Apr 2023 (v1), last revised 28 Apr 2023 (v2)


HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License


Concurrent Realizability on Conjunctive Structures

Emmanuel Beffara ✉ 

LIG, Université Grenoble-Alpes, France

Félix Castro ✉ 

IRIF, Université Paris-Cité, France

Mauricio Guillermo ✉ 

IMERL, Universidad de la República, Uruguay

Étienne Miquey ✉ 

I2M, Université Aix-Marseille, France

Abstract

This work aims at exploring the algebraic structure of concurrent processes and their behavior independently of a particular formalism used to define them. We propose a new algebraic structure called conjunctive involutive monoidal algebra (CIMA) as a basis for an algebraic presentation of concurrent realizability, following ideas of the *algebrization program* already developed in the realm of classical and intuitionistic realizability [5, 14, 15]. In particular, we show how any CIMA provides a sound interpretation of multiplicative linear logic MLL. This new structure involves, in addition to the tensor \otimes and the orthogonal map $(\cdot)^\perp$, a parallel composition $|$. We define the canonical model of this structure as induced by a variant of the π -calculus with global fusions. Using a model of terms, we prove that the parallel composition cannot be defined from the conjunctive structure alone.

1 Introduction

Realizability and its algebrization. Realizability provides a well-established and general set of techniques for studying the relationships between programs and proofs. In the traditional presentation of intuitionistic realizability, one starts from a set A of realizers, which are objects with computational meaning, programs in some formalism (codes of recursive functions, λ -terms, etc). Logic is then interpreted in the powerset of A , in such a way that the meaning of a formula is essentially a set of programs sharing a computational behavior dictated by the formula. From an algebraic viewpoint, the set A induces in its powerset a *Heyting algebra*, which in turn induces a *topos* [11, 19].

Classical realizability adapts these principles to classical logic, building on different foundations [12]. The computational part is based on the duality between programs (potential proofs) and environments (counter-proofs). The proper categorical structure underlying classical realizability was discovered by Streicher [21] and involves an *ordered combinatory algebra* (OCA) induced by terms and stacks. This construction was later generalized in a different setting, namely Krivine ordered combinatory algebras [5], with the feature that Streicher’s construction can be carried out in a purely axiomatic context. In particular, both truth values and realizers belong to the underlying set of these structures and the partial order subsumes subtyping, term reduction and the realizability relation. A similar approach was later followed within Miquel’s implicative algebras [14] (and some variants by the last author’s [15], a slightly different structure which allows to encompass intuitionistic and classical models of forcing and realizability into a general framework, keeping the feature that truth values and realizers (and also forcing conditions) both belong to the underlying set of the IA. All these works therefore advocate for foundations of realizability seeking to transform a rather complex and operational definition into a much simpler and algebraic one.

Logic for concurrency. Process calculi form a wide range of formalisms designed to model concurrent systems and reason about them by means of term rewriting. Their applications

47 are diverse, from the semantics of proof systems to the conception of concrete programming
 48 languages. Type systems for such calculi are therefore a wide domain, with systems of
 49 different kinds designed to capture different behaviors and ensure different properties of
 50 processes: basic interfacing, input-output discipline, linearity, lock-freeness, termination,
 51 respect of communication protocols, etc. To better understand the diversity of calculi and
 52 uncover basic structures and general patterns, many authors have searched for languages with
 53 simpler or more general theory in which the most features could be expressed by means of
 54 restrictions or encodings. Similar unification has been searched for in the realm of types, but
 55 no system can yet claim to be as basic and universal as, *e.g.*, simple types for the λ -calculus,
 56 which perfectly capture the abstraction and application mechanism in the logical world.

57 Concurrent realizability, developed by Beffara [1], transposes the ideas of classical real-
 58 izability in a context of interacting concurrent processes. Following the constructions of
 59 phase semantics from linear logic [7] and ludics [8], the combinatory algebra is replaced
 60 with a variant of the π -calculus, endowed with a structure of Abelian monoid under parallel
 61 composition. The pole, which defines orthogonality as in classical realizability, is seen as a
 62 testing protocol between processes, and the sets of processes that are closed by biorthogonality
 63 act both as truth values and behaviors. However, in the original construction, processes are
 64 subject to polarization and interfacing constraints which restrict the use of names, in order
 65 to avoid ambiguity when composing and ensure consistency. As a consequence, operations
 66 over behaviors inherit such restrictions, which makes them partial in general.

67 The aim of the present work is to set the bases for an algebraic presentation of concurrent
 68 realizability, as a way to study processes and their types in a well-structured algebraic
 69 framework. We propose that, as with sequential models, the algebraic presentation is based
 70 on an ordered algebraic structure which must allow interpreting truth values and concurrent
 71 programs in its underlying set, subsuming the reduction semantics in the order relation.
 72 Furthermore, we want to avoid imposing a priori restrictions on processes.

73 **Principles of the construction.** We aim for an algebraic structure in which we can
 74 represent processes, types and operators in a uniform setting and we will refer to elements
 75 of such a structure as *behaviors*. A guiding intuition is to see behaviors as particular sets
 76 of terms in a process algebra, with some notion of closure by observational equivalence (a
 77 formal implementation is presented in Section 4). We postulate three fundamental structures
 78 over behaviors:

- 79 ■ a complete lattice structure, taking a comparison $a \leq b$ to mean that b exhibits more
 80 possible behaviors than a ;
- 81 ■ a binary operator \otimes , continuous with respect to the lattice structure, that represents an
 82 operation of parallel composition without interaction.
- 83 ■ an anti-monotone unary operator $(\cdot)^\perp$ s.t. a^\perp represents the tests that a passes.

84 This yields a variant of the last author's conjunctive structure [15], including notions of arrow
 85 and application and allowing for a sound interpretation of multiplicative linear logic.

86 To account for concurrency, we decompose the general parallel composition into composi-
 87 tions without interaction—the tensor—and connections that create synchronisation, using a
 88 notion of fusions. Last but not least, we will ensure that names are not a primitive notion of
 89 the structures we define.

90 **Outline of the paper.** We start by introducing in Section 2 the conjunctive structures
 91 which we identified to be conducive to the analysis of concurrent computations from an
 92 algebraic viewpoint. We then develop on an example of concurrent calculus in Section 3,
 93 processes with generalized fusions, on top of which we build a realizability model in Section 4

94 with the expected algebraic structure. We then explain in Section 5 how to equip conjunctive
 95 structures with parallelism, and we prove in Section 6 that the extra structure for this is
 96 independent from the axiomatic of conjunctive structures.

97 **2 Logic in conjunctive structures**

98 Following the program of algebrization of realizability models that was mostly undertaken
 99 in the realm of Krivine realizability [21, 5, 14, 15] and previous work by the first author
 100 regarding concurrent computation [1], we introduce a particular class of *conjunctive algebras*
 101 which we identify as the key algebraic structures underlying realizability models induced by
 102 process calculi. We first define the notion of *conjunctive structure*, which reflect the algebraic
 103 structure of these models (*how are the truth values defined?*), and then define a notion of
 104 *separator* that allows to capture the logical content (*which processes define valid realizers?*).

105 **2.1 Conjunctive structures**

106 ► **Definition 1.** A *conjunctive structure (CS)* is a tuple $(\mathbb{C}, \preceq, \otimes, (\cdot)^\perp)$ such that

- 107 1. (\mathbb{C}, \preceq) is a complete lattice;
- 108 2. \otimes is a binary monotone operation of \mathbb{C} and $(\cdot)^\perp$ is a unary antimonotone function on \mathbb{C} ;
- 109 3. the join operation \bigvee is distributive over \otimes , i.e. for any $a \in \mathbb{C}$ and $\mathfrak{B} \subseteq \mathbb{C}$ we have
 110 $\bigvee_{b \in \mathfrak{B}} (a \otimes b) = a \otimes (\bigvee_{b \in \mathfrak{B}} b)$ and $\bigvee_{b \in \mathfrak{B}} (b \otimes a) = (\bigvee_{b \in \mathfrak{B}} b) \otimes a$;
- 111 4. the orthogonal map $(\cdot)^\perp$ satisfies De Morgan's law $(\bigvee_{b \in \mathfrak{B}} b)^\perp = \bigwedge_{b \in \mathfrak{B}} b^\perp$.

112 We say that the structure is involutive (CIS) if $(\cdot)^\perp$ is involutive: $a^{\perp\perp} = a$ for all $a \in \mathbb{C}$.
 113 Also we call unitary every CS (resp. CIS) with a distinguished element $1 \in \mathbb{C}$.

114 The draw the comparison with the conjunctive structures defined in the last author's
 115 work [15], the only difference is lies in the use of an orthogonal map instead of a negation \neg
 116 that was meant to convey a computational content. In particular, even in the classical case,
 117 a and $\neg\neg a$ are only logically equivalent in [15], not equal.

118 ► **Example 2.** Any complete Boolean algebra $(\mathbb{B}, \preceq, \wedge, \vee, \neg)$ defines a CIS using the conjunc-
 119 tion as tensor $(a \otimes b \triangleq a \wedge b)$ and the negation for orthogonal map $(a^\perp \triangleq \neg a)$.

120 ► **Example 3.** A phase space [7] is defined by an abelian monoid M and a subset $\perp \subseteq M$.
 121 For $a \subseteq M$, define the dual $a^\perp \triangleq \{x : \forall y \in a, xy \in \perp\}$; define $a \otimes b \triangleq \{xy : x \in a, y \in b\}^{\perp\perp}$
 122 for $a, b \subseteq M$. Then the set of subsets $a \subseteq M$ such that $a^{\perp\perp} = a$ forms a CIS.

123 Given a conjunctive structure $(\mathbb{C}, \preceq, \otimes, (\cdot)^\perp)$, we define the usual connectives and quantifiers
 124 of MLL on \mathbb{C} , where F is a function over \mathbb{C} :

$$125 \quad a \wp b \triangleq (a^\perp \otimes b^\perp)^\perp \quad a \multimap b \triangleq (a \otimes b^\perp)^\perp \quad \exists F \triangleq \bigvee_{a \in \mathbb{C}} F(a) \quad \forall F \triangleq \bigwedge_{a \in \mathbb{C}} F(a)$$

126 These definitions induces a canonical interpretation of MLL formulas within any CIS.

127 ► **Definition 4.** Consider a unitary CIS $(\mathbb{C}, \preceq, \otimes, (\cdot)^\perp, 1)$. The interpretation of MLL
 128 formulas with parameters $P \in \mathbb{C}$ is defined as follows:

$$129 \quad \begin{array}{l|l|l|l} \llbracket \mathbb{1} \rrbracket \triangleq 1 & \llbracket P \rrbracket \triangleq P & \llbracket A \otimes B \rrbracket \triangleq \llbracket A \rrbracket \otimes \llbracket B \rrbracket & \llbracket \exists X.A \rrbracket \triangleq \exists (P \mapsto \llbracket A \rrbracket \{X := P\}) \\ \llbracket \perp \rrbracket \triangleq 1^\perp & \llbracket P \rrbracket \triangleq P & \llbracket A \wp B \rrbracket \triangleq \llbracket A \rrbracket \wp \llbracket B \rrbracket & \llbracket \forall X.A \rrbracket \triangleq \exists (P \mapsto \llbracket A \rrbracket \{X := P\}) \end{array}$$

130 Given a sequent $\vdash A_1, \dots, A_k$, the interpretation $\llbracket A_1, \dots, A_k \rrbracket$ is defined as $\llbracket A_1 \rrbracket$ if $k = 1$ and
 131 $\llbracket A_1 \rrbracket \wp \llbracket A_2, \dots, A_k \rrbracket$ otherwise.

$$\begin{array}{c}
\frac{}{\vdash \mathbf{1} : \mathbb{1}}^{(\mathbb{1})} \quad \frac{}{\vdash \mathbf{I} : A^\perp, A}^{(\text{Ax})} \quad \frac{\vdash a : A_1, \dots, A_k}{\vdash \mathbf{ex}(\sigma) * a : A_{\sigma(1)}, \dots, A_{\sigma(k)}}^{(\text{ex})} \quad \frac{\vdash a : \Gamma, A \quad \vdash b : B, \Delta}{\vdash \mathbf{t} * a * b : \Gamma, A \otimes B, \Delta}^{(\otimes)} \\
\frac{\vdash a : \Gamma, A \quad \vdash b : A^\perp, \Delta}{\vdash \mathbf{c} * a * b : \Gamma, \Delta}^{(\text{CUT})} \quad \frac{\vdash a : \Gamma, A\{X := B\}}{\vdash a : \Gamma, \exists X.A}^{(\exists)} \quad \frac{\vdash a : \Gamma, A \quad X \text{ not free in } \Gamma}{\vdash a : \Gamma, \forall X.A}^{(\forall)}
\end{array}$$

■ **Figure 1** Semantic typing rules for MLL

132 ▶ **Remark 5.** As usual, the interpretation $\llbracket A \rrbracket$ of a formula A depends only upon the assignment
133 of the free variables of A . The same is valid for sequents.

134 ▶ **Remark 6.** If \mathbb{C} is a CIS, the involutivity of $(\cdot)^\perp$ actually implies that $(\mathbb{C}, \preceq, \multimap)$ is an
135 implicative structure in the sense of Miquel [14] since \multimap satisfies the expected variance and
136 continuity properties: in particular $(\bigvee_{b \in B} b) \multimap a = \bigwedge_{b \in B} (b \multimap a)$ for any $a \in \mathbb{C}$ and $B \subseteq \mathbb{C}$.

137 2.2 Separators and internal logic

138 Conjunctive structures are suited to interpret formulas of MLL. To account for a validity
139 notion, we need to introduce the notion of separator [14, 15], a subset of the structure which
140 intuitively distinguishes valid formulas, similarly to a filter in a Boolean algebra. Separators
141 are built upon a set of combinators that one can understand as axioms for the internal logic.

142 ▶ **Definition 7.** Consider a unitary CS $(\mathbb{C}, \preceq, \gamma, \otimes, (\cdot)^\perp, 1)$. The MLL combinators are:

$$\begin{array}{ll}
\blacksquare S_3 \triangleq \bigwedge_{a, b \in \mathbb{C}} (a \otimes b) \multimap (b \otimes a) & \blacksquare S_6 \triangleq \bigwedge_{a \in \mathbb{C}} a \multimap (1 \otimes a) \\
\blacksquare S_4 \triangleq \bigwedge_{a, b, c \in \mathbb{C}} (a \multimap b) \multimap (a \otimes c) \multimap (b \otimes c) & \blacksquare S_7 \triangleq \bigwedge_{a \in \mathbb{C}} (1 \otimes a) \multimap a \\
\blacksquare S_5 \triangleq \bigwedge_{a, b, c \in \mathbb{C}} ((a \otimes b) \otimes c) \multimap (a \otimes (b \otimes c)) & \blacksquare S_8 \triangleq \bigwedge_{a, b \in \mathbb{C}} (a \multimap b) \multimap (b^\perp \multimap a^\perp)
\end{array}$$

144 The definition (and name) for the combinators S_3, S_4, S_5 come from the notion of separator
145 in [15], in which we replaced here S_1 and S_2 , that were not linear, by S_6, S_7, S_8 .

146 ▶ **Definition 8.** Let us consider $(\mathbb{C}, \preceq, \gamma, \otimes, (\cdot)^\perp, 1)$ a unitary CIS. A monoidal separator
147 on \mathbb{C} is an upwards closed set $\mathcal{S} \subseteq \mathbb{C}$ such that:

- 148 ■ $\mathcal{A} \subseteq \mathcal{S}$, and for any $a, b \in \mathbb{C}$, if $a \multimap b \in \mathcal{S}$ and $a \in \mathcal{S}$ then $b \in \mathcal{S}$.
- 149 ■ \mathcal{S} contains the MLL-combinators S_3, \dots, S_8 .

150 A conjunctive involutive monoidal algebra (CIMA) is a (unitary) CIS \mathbb{C} together with a
151 monoidal separator \mathcal{S} .

152 ▶ **Example 9.** In any CIS induced by a complete Boolean algebra \mathbb{B} (see Example 2), all the
153 combinators are tautologies trivially interpreted by the maximal element \top . Therefore, the
154 singleton $\{\top\}$ (or alternatively any filter on \mathbb{B}) defines a separator for this CIS.

155 Let \mathbb{C} be a CIMA. We call *semantic judgement* a statement $\vdash a : \Gamma$ where the “term” a is an
156 element $a \in \mathbb{C}$ and $\vdash \Gamma$ is a sequent of formulas with with parameters in \mathbb{C} . Such a judgement
157 is *sound* if $a \preceq \llbracket \Gamma \rrbracket$. A *semantic typing rule* is an inference rule where the premises and
158 conclusion are semantic judgements, possibly involving free variables. Such a rule is *sound* if
159 the soundness of its premises entails the soundness of its conclusion for any instantiation of
160 the variables in \mathbb{C} .

161 Using the following terms (where $\sigma : [1; k] \rightarrow [1; k]$ is a permutation):

$$\begin{array}{l}
\mathbf{I} \triangleq \bigwedge_{a \in \mathbb{C}} (a \multimap a) \\
\mathbf{t} \triangleq \bigwedge_{a, b, g, d \in \mathbb{C}} ((g \wp a) \multimap (b \wp d) \multimap g \wp ((a \otimes b) \wp d)) \\
\mathbf{c} \triangleq \bigwedge_{a, g, d \in \mathbb{C}} ((g \wp a) \multimap (a^\perp \wp d) \multimap (g \wp d)) \\
\mathbf{ex}(\sigma) \triangleq \bigwedge_{a_1, \dots, a_k \in \mathbb{C}} ((a_1 \wp \dots \wp a_k) \multimap (a_{\sigma(1)} \wp \dots \wp a_{\sigma(k)}))
\end{array}$$

163 we provide semantic typing rules for MLL in Figure 1, where $a * b \triangleq \lambda \{c \in \mathbb{C} \mid a \preceq b \multimap c\}$
 164 is the usual application from implicative algebras.

165 ► **Theorem 10.** *The semantic typing rules of Figure 1 are sound for all CIMA.*

166 **Proof.** The proof consists in proving that for any CIMA $(\mathbb{C}, \mathcal{S})$ and any permutation σ , the
 167 combinators \mathbf{I} , \mathbf{t} , \mathbf{c} and $\text{ex}(\sigma)$ belong to the separator \mathcal{S} , see Appendix A. ◀

168 3 Processes with global fusions

169 In this section, we define a computational structure that will serve as our reference. We first
 170 recall the basic definitions of a standard π -calculus, then we build processes with fusions as
 171 a formal extension of this language. These processes will serve as realizers later on.

172 The choice of a particular variant of the π -calculus is essentially arbitrary, we pick this
 173 one for the purpose of illustration. The only features we need are the ability to perform
 174 renaming and the availability of parallel composition and hiding with usual properties. The
 175 point of the construction is to extend the language with name fusions at top-level while
 176 preserving this composition structure. This allows for working with fusions in the algebraic
 177 structure without imposing constraints on the underlying process language.

178 3.1 Basic processes

179 The following recalls the standard definitions of the polyadic asynchronous π -calculus with
 180 its operational semantics by reduction [20]. We leave out replication and addition because
 181 these will not be used in this paper, but including them is not a problem.

182 ► **Definition 11.** *The set Π of π -processes is defined by the following grammar:*

$$183 \quad P, Q ::= \mathbf{1} \mid (P \mid Q) \mid u(\vec{x}).P \mid \bar{u}(\vec{v}) \mid (\nu y)P$$

184 where \vec{x} is a finite sequence of pairwise distinct names, \vec{v} is a finite sequence of names, y and u
 185 are names. An expression $u(\vec{x})$ is called reception, an expression $\bar{u}(\vec{v})$ is called emission.

186 The names in \vec{x} are bound in $u(\vec{x}).P$, the name y is bound in $(\nu y)P$. The set of free
 187 names of a term P is denoted by $\text{FN}(P)$. Terms are considered up to renaming of bound
 188 names, also called α -conversion, written \equiv_α .

189 Structural equivalence is the smallest congruence over terms such that

- 190 ■ (Π, \mid, \equiv) is an abelian monoid, i.e. $\mathbf{1} \mid P \equiv P$, $P \mid Q \equiv Q \mid P$ and $(P \mid Q) \mid R \equiv P \mid (Q \mid R)$,
- 191 ■ $P \equiv (\nu x)P$ and $P \mid (\nu x)Q \equiv (\nu x)(P \mid Q)$ whenever $x \notin \text{FN}(P)$,
- 192 ■ $(\nu x)(\nu y)P \equiv (\nu y)(\nu x)P$ and we write $(\nu xy)P \triangleq (\nu x)(\nu y)P$.

193 ► **Definition 12.** A substitution is a function $\sigma : \mathbb{N} \rightarrow \mathbb{N}$. Provided that the names in \vec{x} are
 194 pairwise distinct and that \vec{x} and \vec{v} have the same length, $\{\vec{x} := \vec{v}\}$ denotes the substitution
 195 which replaces \vec{x} by \vec{v} and leaves all other names unchanged.

196 Given $X \subseteq \mathbb{N}$ we denote by $\sigma \upharpoonright_X$ the substitution which coincides with σ on X and is the
 197 identity outside. We write $\sigma \setminus X$ for the substitution $\sigma \upharpoonright_{X^c}$.

198 We denote by P^σ the result of applying σ to $P \in \Pi$, defined inductively in the standard
 199 way. Given substitutions $\sigma_1, \dots, \sigma_k$ we denote by $P^{\sigma_1 \dots \sigma_k}$ the term $(\dots (P^{\sigma_1}) \dots)^{\sigma_k}$.

200 ► **Definition 13.** One-step reduction is the smallest binary relation \longrightarrow_1 over Π such that

- 201 ■ $\bar{u}(\vec{v}) \mid u(\vec{x}).P \longrightarrow_1 P\{\vec{x} := \vec{v}\}$,
- 202 ■ \longrightarrow_1 is compatible with \equiv , i.e. if $P \equiv P'$, $Q \equiv Q'$ and $P \longrightarrow_1 Q$ then $P' \longrightarrow_1 Q'$,

203 ■ \longrightarrow_1 is compatible with parallel composition and hiding, i.e. if $x \in \mathbb{N}$ and $P \longrightarrow_1 P'$,
 204 then $P \mid Q \longrightarrow_1 P' \mid Q$ and $(\nu x)P \longrightarrow_1 (\nu x)P'$.

205 We do not define a higher-level semantics at this point. The semantics induced by the
 206 conjunctive structures defined in the following sections will be a generic form of testing and
 207 most of the construction will be parametric in the particular testing protocol.

208 3.2 Processes with fusions

209 A *fusion* is, intuitively, a connection between channels. The defining property of a fusion
 210 $u \leftrightarrow v$ is that its presence within a process allows actions on channel u to synchronize with
 211 actions on channel v . Several works have defined extensions of the π -calculus with this
 212 feature [6, 18, 22] and studied its effects on the theory and the expressiveness of the language.

213 Instead, the construction presented in this section rather aims at extending an arbitrary
 214 process calculus with fusions *without* affecting its theory, for our study to be compatible
 215 with any particular process language. This is made possible by the fact that we only need
 216 fusions at top-level, so that they can live side-by-side with standard processes. Intuitively, a
 217 fusion $u \leftrightarrow v$ at top level in a process can be interpreted as the information that u and v will
 218 eventually get substituted with the same name.

219 ► **Definition 14.** A fusion is an equivalence relation over \mathbb{N} . The set of fusions is written \mathcal{E} .
 220 Given $e \in \mathcal{E}$ we write $x \sim_e y$ instead of $(x, y) \in e$. For $e, f \in \mathcal{E}$ we write ef for the smallest
 221 equivalence that contains both e and f .

222 ► **Remark 15.** The set \mathcal{E} is a complete lattice under inclusion, with the identity $\Delta_{\mathbb{N}} \triangleq$
 223 $\{(x, x) \mid x \in \mathbb{N}\}$ as minimum and the full relation $\nabla_{\mathbb{N}} \triangleq \mathbb{N} \times \mathbb{N}$ as maximum. The meet of a
 224 set $S \subseteq \mathcal{E}$ is its intersection $\bigcap S$ and its join is the transitive closure of its union $\bigcup S$.

225 ► **Definition 16.** Given $e \in \mathcal{E}$, $x, y \in \mathbb{N}$, $X \subseteq \mathbb{N}$ and $\sigma : \mathbb{N} \rightarrow \mathbb{N}$, define

- 226 ■ the equivalence class of x as $[x]_e \triangleq \{y \in \mathbb{N} \mid x \sim_e y\}$, extended as $[X]_e \triangleq \bigcup_{x \in X} [x]_e$;
- 227 ■ the domain of e as $|e| \triangleq \{x \in \mathbb{N} \mid [x]_e \neq \{x\}\}$;
- 228 ■ the restriction of e to X as $e \cap X \triangleq (e \cap (X \times X)) \cup \Delta_{\mathbb{N}}$;
- 229 ■ the hiding of X in e as $e \setminus X \triangleq e \cap X^c$;
- 230 ■ the elementary fusion of x and y as $x \leftrightarrow y \triangleq \{(x, y), (y, x)\} \cup \Delta_{\mathbb{N}}$.

231 An equivalence e can be induced by a function σ , considering that equivalent elements are
 232 the ones with the same image under σ . Such a function can always be deduced from e by
 233 picking a representative in each equivalence class. The following definitions formalize these
 234 constructions, exploiting the well-ordering over \mathbb{N} to get canonical representatives.

235 ► **Definition 17.** Given a fusion $e \in \mathcal{E}$, $x \in \mathbb{N}$ and a function $\tau : \mathbb{N} \rightarrow \mathbb{N}$, we define:

- 236 ■ $x_e^\bullet \triangleq \min[x]_e$ and $x_e^* \triangleq \begin{cases} x & \text{if } [x]_e = \{x\} \\ \min([x]_e \setminus \{x\}) & \text{if } [x]_e \neq \{x\} \end{cases}$
- 237 ■ $\sigma_e^\bullet : x \mapsto x_e^\bullet$ for all x , the substitution induced by e ;
- 238 ■ $\varepsilon_\tau \triangleq \bigvee_{x \in \mathbb{N}} (x \leftrightarrow \tau(x))$, the fusion induced by τ ;
- 239 ■ $e^\tau \triangleq \bigvee_{x \sim_e y} (\tau(x) \leftrightarrow \tau(y))$, the composition of e with τ .

240 We are now ready to introduce the notion of processes with global fusions, and extend
 241 the construction on processes to this generalization. This is similar to that of unitization in
 242 algebras, where a non-unital algebra is extended with a formal extra element and saturated
 243 to preserve the structure and endow the extra element with desired properties.

244 ▶ **Definition 18.** *The set of processes with global fusions (PGF) is defined as*

$$245 \quad \bar{\Pi} \triangleq \Pi \times \mathcal{E} = \{(P, e) \mid P \in \Pi, e \in \mathcal{E}\}$$

246 *Over $\bar{\Pi}$, define substitution, free names, α -equivalence and structural equivalence \equiv as*

$$247 \quad (P, e)^\tau \triangleq (P^\tau, e^\tau) \quad (P, e) \equiv_\alpha (Q, f) \iff e = f \text{ and } P^{\sigma_e} \equiv_\alpha Q^{\sigma_f}$$

$$248 \quad \text{FN}(P, e) \triangleq \text{FN}(P) \cup |e| \quad (P, e) \equiv (Q, f) \iff e = f \text{ and } P^{\sigma_e} \equiv Q^{\sigma_f}$$

250 When it is not necessary to specify process and fusion, we will use the lowercase letters p, q, r, s
251 to refer to processes with fusions.

252 Remark that the definition α -equivalence and structural equivalence allows renaming of
253 free names as long as the equivalence class of each name is unchanged, so that we have for
254 instance $(a(x).P, a \leftrightarrow b) \equiv (b(x).P, a \leftrightarrow b)$. Including $|e|$ in $\text{FN}(P, e)$ ensures that, despite such
255 renaming, FN is invariant under \equiv_α as expected. Note also that if we identify each term P
256 with the pair $(P, \Delta_{\mathbb{N}})$, then the operations defined above coincide with those of Π .

257 ▶ **Definition 19.** *One-step reduction in $\bar{\Pi}$ is the relation \longrightarrow_1 such that $(P, e) \longrightarrow_1 (Q, f)$ if
258 and only if $e = f$ and $P^{\sigma_e} \longrightarrow_1 Q^{\sigma_f}$. Multistep reduction \longrightarrow is the reflexive and transitive
259 closure of \longrightarrow_1 .*

260 Again, this extends the associated relation from Π to $\bar{\Pi}$ and allows more reductions
261 when the fusion relates two names on which compatible actions occur. For instance we have
262 $(u(x).P \mid \bar{v}(y), u \leftrightarrow v) \longrightarrow_1 (P\{x := y\}, u \leftrightarrow v)$.

263 ▶ **Definition 20.** *For $(P, e), (Q, f) \in \bar{\Pi}$, define $(P, e) \mid (Q, f) \triangleq (P \mid Q, ef)$.*

264 Identifying a pure fusion e with the pair $(\mathbb{1}, e)$, we have $(P, e) \equiv P \mid e$ for every P , since
265 $P \mid \mathbb{1} \equiv P$ in Π . So in this sense Π and \mathcal{E} generate $\bar{\Pi}$ under parallel composition. Besides,
266 reduction \longrightarrow_1 is compatible with \equiv and parallel composition, as in the basic calculus.

267 We now need to extend name hiding to $\bar{\Pi}$. The result of $(\nu x)(P, e)$ must be (Q, f)
268 where $x \notin \text{FN}(Q, f)$, hence $x \notin \text{FN}(Q) \cup |f|$. Consider a pair $(\nu x)(\bar{x}, x \leftrightarrow y)$: even if x
269 must be bound in $(\nu x)(\bar{x}, x \leftrightarrow y)$, since x is fused with y , we have $(\bar{x}, x \leftrightarrow y) \equiv_\alpha (\bar{y}, x \leftrightarrow y)$ so
270 the only reasonable definition for $(\nu x)(\bar{x}, x \leftrightarrow y)$ is $(\bar{y}, \Delta_{\mathbb{N}})$, which is structurally equivalent
271 to $((\nu x)\bar{y}, \Delta_{\mathbb{N}})$. On the other hand, if we consider $(\nu x)(\bar{x}, \Delta_{\mathbb{N}})$, we expect to get simply
272 $((\nu x)\bar{x}, \Delta_{\mathbb{N}})$. Definition 17 provides the notation x_e^* to represent this in a general way.

273 ▶ **Definition 21.** *For $(P, e) \in \bar{\Pi}$ and $x \in \mathbb{N}$, define $(\nu x)(P, e) \triangleq ((\nu x)P\{x := x_e^*\}, e \setminus \{x\})$.*

274 ▶ **Remark 22.** Whenever x is not equivalent to another name, *i.e.* $[x]_e = \{x\}$, we have $x = x_e^*$
275 and therefore the binder νx on a PGF process only computes as expected on the process
276 side: $(\nu x)(P, e) = ((\nu x)P, e)$. If, instead, x is equivalent to other names, *i.e.* $[x]_e \neq \{x\}$, then
277 $x \neq x_e^*$ and νx will disconnect x from its fusion-side equivalents, while replacing it by an
278 equivalent name on the process side: $((\nu x)P\{x := x_e^*\}, e \setminus \{x\}) \equiv (P\{x := x_e^*\}, e \setminus \{x\})$.

279 To extend the definition of ν to finite sets, we first need to ensure that, up to α -equivalence,
280 the order of application of ν is irrelevant.

281 ▶ **Lemma 23.** *Consider $(P, e) \in \bar{\Pi}$ and $x, y \in \mathbb{N}$, then $(\nu x)(\nu y)(P, e) \equiv_\alpha (\nu y)(\nu x)(P, e)$.*

282 **Proof.** See Appendix B. ◀

283 ▶ **Definition 24.** *Given a finite set of names $X = \{x_1, \dots, x_k\}$ with $x_1 < \dots < x_k$, define*
284 $(\nu X)(P, e) \triangleq (\nu x_k) \cdots (\nu x_1)(P, e)$.

285 Later on, we will need to extend this definition for a possibly infinite set X . Even if,
 286 for any PGF p , there exists only finitely many free names of p in X which we need to hide,
 287 for each of them we need to treat them differently depending on whether their equivalence
 288 classes is included in X or not. The following definitions generalize the unary definition of ν
 289 in that regards.

290 ► **Definition 25.** Consider a set $X \subseteq \mathbb{N}$ and a fusion $e \in \mathcal{E}$. Define the substitution

$$291 \quad \mu_{(X,e)}(x) \triangleq \begin{cases} x_e^\bullet & \text{if } [x]_e \subseteq X \\ \min([x]_e \setminus X) & \text{if } [x]_e \setminus X \neq \emptyset \end{cases}$$

292 ► **Definition 26.** Let $(P, e) \in \bar{\Pi}$ and $X \subseteq \mathbb{N}$. Since $\text{FN}(P)$ is finite, there exists classes
 293 $[x_1]_e, \dots, [x_{h+k}]_e$ in \mathbb{N}/e such that for each $i \leq h$ we have $[x_i]_e \subseteq X$ and for each $j > h$ we
 294 have $[x_j]_e \setminus X \neq \emptyset$, and moreover $\text{FN}(P) \cap X \subseteq \bigcup_{i=1}^{h+k} [x_i]_e$. Define

$$295 \quad (\nu X)(P, e) \triangleq ((\nu x_{1,e}^\bullet, \dots, x_{h,e}^\bullet)P^{\mu_{(X,e)}}, e \setminus X)$$

296 Let us denote by $\bar{\nu}$ the binder $(\nu \mathbb{N})$.

297 4 Conjunctive structure of processes

298 Following the first author's work [1], we shall now see how to define realizability models which
 299 interpret MLL formulas as sets of processes. This models actually induces a conjunctive
 300 involutive monoidal algebra and as such provides a sound interpretation of MLL. In particular,
 301 we would like to emphasize that this illustrate a significant benefit of our methodology, which
 302 it allows us to isolate logical properties that are shared by *all* concurrent realizability models
 303 sharing this algebraic structure, ensuring that these properties are indeed insensitive to
 304 implementation details such as the choice of a particular syntax for processes.

305 4.1 Concurrent realizability with PGF

306 We build upon the main insight of Krivine's realizability [13], by parameterizing the interpret-
 307 ation by a *pole*, a set of processes somewhat characterizing valid interactions between terms.
 308 In this setting, we will consider poles that are barely closed under structural equivalence to
 309 characterize the soundness of the interaction with respect to the renaming process necessary
 310 to the pure parallel composition, without focusing so far on the reduction of processes. From
 311 now on, we denote by \mathbb{P} the set of all processes $\mathbb{P} \triangleq \mathcal{P}(\bar{\Pi})$.

312 ► **Definition 27.** A pole is a set of closed processes $\perp \subseteq \{p \in \bar{\Pi} : \text{FN}(p) = \emptyset\}$ that is closed
 313 under \equiv , i.e. if $p \equiv q$ and $q \in \perp$ then $p \in \perp$.

314 Given a pole \perp , we say that two processes p, q are *orthogonal*, which we write $p \perp q$,
 315 whenever $\bar{\nu}(p \mid q) \in \perp$. This naturally induces an orthogonal operator $(\cdot)^\perp : \mathbb{P} \rightarrow \mathbb{P}$ on set of
 316 processes by defining $A^\perp \triangleq \{p \in \bar{\Pi} : \forall q \in A. p \perp q\}$. This operator satisfies the usual property
 317 of orthogonality.

318 ► **Proposition 28.** For all $A, B \in \mathbb{P}$ and for any non-empty $\mathfrak{B} \subseteq \mathbb{P}$ we have:

319 1. If $A \subseteq B$ then $B^\perp \subseteq A^\perp$. 2. $A \subseteq A^{\perp\perp}$ and $A^\perp = A^{\perp\perp\perp}$. 3. $(\bigcup_{X \in \mathfrak{B}} X)^\perp = \bigcap_{X \in \mathfrak{B}} X^\perp$
 320 and if $\mathfrak{B} \neq \emptyset$ then $(\bigcap_{X \in \mathfrak{B}} X)^\perp \supseteq (\bigcup_{X \in \mathfrak{B}} X^\perp)$ 4. $(\bigcup_{X \in \mathfrak{B}} X^{\perp\perp})^\perp = (\bigcup_{X \in \mathfrak{B}} X)^\perp$.

321 Our realizability interpretation is inspired by the semantics of phases, in particular we
 322 interpret formulas in the set of *behaviors* $\mathbb{B} \triangleq \{A \in \mathbb{P} : A^{\perp\perp} = A\}$, i.e. the sets that are
 323 closed under double orthogonal. We first show that this set can be endowed with a structure
 324 of complete lattice.

325 ▶ **Proposition 29.** *Let us define $\mathbb{V} \mathfrak{B} \triangleq (\bigcup \mathfrak{B})^{\perp\perp}$ for all $\mathfrak{B} \subseteq \mathbb{B}$. Then:*

- 326 1. $(\mathbb{P}, \bigcap, \bigcup, \subseteq)$ is a complete lattice with top element $\overline{\mathbb{I}}$ and bottom element \emptyset .
 327 2. $(\mathbb{B}, \bigcap, \bigvee, \subseteq)$ is a complete lattice with top element $\overline{\mathbb{I}}$ and bottom element $\emptyset^{\perp\perp}$.

328 In order to interpret MLL formulas in this framework, the main idea consists in seeing
 329 the tensor \otimes as defining the parallel composition of two processes in such a way that they
 330 do not communicate. This is achieved by making extensive use of substitutions to assign
 331 each process a disjoint space of names. Technically, we rely on the following the fact that \mathbb{N}
 332 is in bijection with the set of even (or odd) natural numbers, somehow reflecting the tree
 333 structure of the formulas into the space of names.

334 ▶ **Definition 30.** *We fix injections $\iota_1, \iota_2 : \mathbb{N} \rightarrow \mathbb{N}$ such that¹ $\iota_1(\mathbb{N}) \cap \iota_2(\mathbb{N}) = \emptyset$ and
 335 $\iota_1(\mathbb{N}) \cup \iota_2(\mathbb{N}) = \mathbb{N}$. We use the following notations: $\mathbb{N}^i \triangleq \iota_i(\mathbb{N})$, $n.i := \iota_i(n)$, $p^i \triangleq p^{\iota_i}$ and
 336 $e^i \triangleq e^{\iota_i}$. Given $p \in \overline{\mathbb{I}}$ s.t. $\text{FN}(p) \subseteq \mathbb{N}^i$ we denote as p^{-i} the process $p^{\iota_i^{-1}}$. Similarly, if e is a
 337 fusion s.t. $|e| \subseteq \mathbb{N}^i$, we denote as e^{-i} the fusion $e^{\iota_i^{-1}}$.*

338 We can now define the semantic counterpart of MLL connectives on processes, which
 339 can then lift to behaviors. For technical purposes, we also define an operator $\cdot * \cdot$, which in
 340 fact coincides with the usual application, i.e. the left adjoint of the linear implication \multimap as
 341 Proposition 34 shows.

342 ▶ **Definition 31.** *For all $p = (P, e)$ and $q = (Q, f)$ define the following operations²:*
 343 $\bullet p \bullet q \triangleq p^1 \mid q^2 = (P^1 \mid Q^2, e^1 f^2) \quad \multimap p * q \triangleq ((\nu \mathbb{N}^1)(p \mid q^1))^{-2} = ((\nu \mathbb{N}^1)(P \mid Q^1, e f^1))^{-2}$.

344 ▶ **Definition 32.** *We define the following operations on \mathbb{P} :*

$$\begin{array}{l|l}
 1 & := \{(\mathbb{1}, \Delta_{\mathbb{N}})\}^{\perp\perp} \\
 A \bullet B & := \{p \bullet q \mid p \in A, q \in B\} \\
 A \parallel B & := (A \mid B)^{\perp\perp} \\
 A * B & := \{p * q \mid p \in A, q \in B\}^{\perp\perp} \\
 \hline
 A \otimes B & := (A \bullet B)^{\perp\perp} \\
 A \wp B & := (A^{\perp} \otimes B^{\perp})^{\perp} \\
 A \multimap B & := (A \otimes B^{\perp})^{\perp} \\
 A \upharpoonright X & := \{p \in A \mid \text{FN}(p) \subseteq X\}
 \end{array}$$

346 Observe that these definitions define behaviors, and besides, they are insensitive to double
 347 orthogonal:

348 ▶ **Proposition 33.** *For any $A, B \in \mathbb{P}$, it holds that $(A \bullet B)^{\perp} = (A^{\perp\perp} \bullet B^{\perp\perp})^{\perp}$. As a
 349 consequence, $A \otimes B = A^{\perp\perp} \otimes B^{\perp\perp}$, $A \wp B = A^{\perp\perp} \wp B^{\perp\perp}$ and $A \multimap B = A^{\perp\perp} \multimap B^{\perp\perp}$.*

350 **Proof.** See Appendix C. ◀

351 ▶ **Proposition 34.** *Let us consider $A, B, C \in \mathbb{P}$. Then:*

- 352 1. $C \subseteq A \multimap B \iff C * A \subseteq B^{\perp\perp}$.
 353 2. $C * A = \bigcap \{B \in \mathbb{B} \mid C \subseteq A \multimap B\} = \min \{B \in \mathbb{B} \mid C \subseteq A \multimap B\}$.

354 (In particular, if $B \in \mathbb{B}$ then $C \subseteq A \multimap B \iff C * A \subseteq B$).

355 **Proof.** See Appendix C. ◀

356 To complete the definition of the concurrent realizability interpretation of MLL using
 357 processes, it only remains to define the realizability relation.

¹ For instance, we can take $\iota_1(n) \triangleq 2n + 1$ and $\iota_2(n) \triangleq 2n$.

² Observe that $\text{FN}((\nu \mathbb{N}^1)((P, e) \mid (Q, f)^1)) \subseteq \mathbb{N}^2$ and thus we can apply ι_2^{-1} to this process.

358 ► **Definition 35.** We say that a PGF p realizes a behavior A , which we write $p \Vdash A$, if $p \in A$.

359 A detailed study of this interpretation and of the computational content of realizers, which
 360 mostly ensure adequate communication between adequate channels using an appropriate
 361 fusion, is out of the scope of this paper. We simply illustrate how the identity can be realized.
 362

363 ► **Lemma 36.** For any $A \in \mathbb{P}$, we have that $I \triangleq \prod_{n \in \mathbb{N}} (n.1 \leftrightarrow n.2) \Vdash A \multimap A$.

364 **Proof.** We have $A \multimap A = (A \bullet A^\perp)^\perp$ (using Proposition 34). Let us consider $p \in A$ and
 365 $q \in A^\perp$, it suffices to prove that $I \perp p^1 \mid q^2$. Using Proposition 63, we get that $\bar{\nu}(p^1 \mid q^2 \mid I) \equiv_\alpha$
 366 $(\nu_2)(\nu_1)(p^1 \mid q^2 \mid I) \equiv_\alpha (\bar{\nu})(p \mid q)$ and we conclude by observing that $(\bar{\nu})(p \mid q) \in \perp$. ◀

367 4.2 The induced conjunctive involutive monoidal algebra

368 Most importantly, we can prove that this construction actually defines a conjunctive involutive
 369 monoidal algebra. In particular, this entails (via Theorem 10) that it defines a valid
 370 interpretation of MLL. The proof is in two parts, we prove that the interpretation induces a
 371 CIS, then we show that the set of realized behaviors defines a valid separator.

372 ► **Proposition 37.** The tuple $(\mathbb{B}, \subseteq, \otimes, (\cdot)^\perp)$, where (\mathbb{B}, \subseteq) is the lattice defined in Proposi-
 373 tion 29, is a conjunctive involutive structure.

374 **Proof.** The variance of the operators \otimes and $(\cdot)^\perp$ is direct by construction, we only have to
 375 verify the distributivity law, see Appendix D. ◀

376 ► **Proposition 38.** The following behaviors are realized by pure fusions:

- | | |
|---|--|
| 1. $\bigcap_{A \in \mathbb{B}} A \multimap A$. | 5. $\bigcap_{A \in \mathbb{B}} A \multimap (1 \otimes A)$. |
| 2. $\bigcap_{A, B \in \mathbb{B}} (A \otimes B) \multimap (B \otimes A)$. | 6. $\bigcap_{A \in \mathbb{B}} (1 \otimes A) \multimap A$. |
| 3. $\bigcap_{A, B, C \in \mathbb{B}} (A \multimap B) \multimap (B \multimap C) \multimap A \multimap C$. | 7. $\bigcap_{A, B \in \mathbb{B}} (A \multimap B) \multimap (B^\perp \multimap A^\perp)$. |
| 4. $\bigcap_{A, B, C \in \mathbb{B}} ((A \otimes B) \otimes C) \multimap (A \otimes (B \otimes C))$. | |

378 **Proof.** The first statement is Lemma 36. The proof of the other statements follows the very
 379 same ideas. For more details, see in appendix. ◀

380 ► **Proposition 39.** The set of non-empty behaviors $\mathcal{S}_{\mathbb{B}} \triangleq \mathbb{B} \setminus \emptyset$ defines a separator for the
 381 conjunctive structure $(\mathbb{B}, \subseteq, \otimes, (\cdot)^\perp)$.

382 **Proof.** $\mathcal{S}_{\mathbb{B}}$ is upwards closed by construction, and the proof that all the combinators are
 383 inhabited by fusions is given in Proposition 38. To prove that it is compatible with the
 384 modus ponens, let $a, b \in \mathbb{B}$ be behaviors and p, q be processes such that $p \in a \multimap b$ and $q \in a$.
 385 By Proposition 34, we get $(a \multimap b) * a \subseteq b$, and thus in particular $p * q \in b$, i.e. $b \in \mathcal{S}_{\mathbb{B}}$. ◀

386 5 Conjunctive algebras for concurrency

387 5.1 Parallel composition

388 So far, we identified the conjunctive part of the realizability model, that is, processes handling
 389 disjoint namespaces. We shall now investigate the necessary structure to algebrize parallelism.

$$\begin{array}{c}
\overline{F(a, \mathbb{1}) \equiv_F a} \quad \overline{F(\mathbb{1}, a) \equiv_F a} \quad \overline{F(a, b) \equiv_F F(b, a)} \quad \overline{F(a, F(b, c)) \equiv_F F(F(a, b), c)} \\
\frac{a \equiv_F a' \quad b \equiv_F b'}{F(a, b) \equiv_F F(a', b')} \quad \frac{a \equiv_F a' \quad b \equiv_F b'}{a \otimes b \equiv_F a' \otimes b'} \quad \frac{a \equiv_F a'}{a^\perp \equiv_F a'^\perp} \quad \frac{a \equiv_F a' \quad b \equiv_F b' \quad a \preceq b}{a' \preceq b'}
\end{array}$$

■ **Figure 2** Axioms for the compositional structure associated to F

390 ▶ **Definition 40.** Let us consider a CS $(\mathbb{C}, \otimes, (\cdot)^\perp, \preceq)$. A composition on \mathbb{C} is an increasing
391 and Υ -continuous function $F : \mathbb{C} \times \mathbb{C} \rightarrow \mathbb{C}$.

392 Given a composition F on \mathbb{C} , define the induced compositional structure associated to F
393 as the quotient \mathbb{C}/\equiv_F where the equivalence relation \equiv_F is the minimum equivalence relation
394 that satisfies the rules of Figure 2.

395 Observe that the first four rules express that $(\mathbb{C}, F, \mathbb{1}, \equiv_F)$ is an abelian monoid while the last
396 ones express that the structure of the CS \mathbb{C} is lifted to the quotient \mathbb{C}/\equiv_F , thus inducing
397 respectively the operations $\otimes_F, (\cdot)_F^\perp$ and the preorder \preceq_F . It is a direct verification that the
398 structure $(\mathbb{C}/\equiv_F, \otimes_F, (\cdot)_F^\perp, \preceq_F)$ is also a CS. Provided there is no ambiguity, we will avoid
399 the subscripts F on the induced operations. The structure $(\mathbb{C}/\equiv_F, \otimes, (\cdot)^\perp, F, \preceq)$ induced by
400 F on \mathbb{C} is denoted as \mathbb{C}_F .

401 ▶ **Definition 41.** Let us consider $(\mathbb{C}, \otimes, (\cdot)^\perp, \mathbb{1}, F, \preceq)$ a CS with composition F . We define
402 $\triangleright_F : \mathbb{C} \times \mathbb{C} \rightarrow \mathbb{C}$ by means of $b \triangleright_F c \triangleq \Upsilon\{x \in \mathbb{C} \mid xFb \preceq c\}$

403 It is straightforward to check that \triangleright_F defines a right adjoint to F (see Appendix E).

404 ▶ **Proposition 42.** Let us consider $(\mathbb{C}, \otimes, (\cdot)^\perp, \mathbb{1}, F, \preceq)$ a CS with composition F . Then
405 $aFb \preceq c \iff a \preceq b \triangleright_F c$ for all $a, b, c \in \mathbb{C}$.

406 ▶ **Proposition 43.** Consider the CS $(\mathbb{B}, \otimes, (\cdot)^\perp, \mathbb{1}, \sqsubseteq)$ of behaviors over $\bar{\Pi}$.

- 407 1. The operation \parallel (from Definition 32) is a composition over \mathbb{C} , the equivalence \equiv_{\parallel} is
408 equality and its right adjoint is $B \triangleright_{\parallel} C := (B \parallel C^\perp)^\perp$.
- 409 2. The operation \otimes (from Definition 32) is a composition over \mathbb{C} and its right adjoint is
410 $B \triangleright_{\otimes} C := (((B^2 \parallel C^\perp)^\perp)_{\downarrow \mathbb{N}^1})^{-1}$.

411 **Proof.** See on the appendix. ◀

412 5.2 Embedding of the π -calculus through Honda-Yoshida's combinators

413 Following previous works on algebraic structures for realizability models based on sequential
414 calculi [14, 15], we face two options to show that the π -calculus can be faithfully embed
415 with conjunctive involutive structure with compositions. Either we find a way to soundly
416 embed the different construct $u(\vec{x}).P, \bar{u}(\vec{v}), |(\nu y)P$, or we can a set of combinators that
417 is complete for this calculus (as are S and K for the λ -calculus). We opt for the second
418 approach, building on Honda & Yoshida combinators for the π -calculus [9].

419 ▶ **Definition 44.** A Honda & Yoshida's structure (HYS) is a CIS $(\mathbb{C}, \preceq, \otimes, |, (\cdot)^\perp, \mathbb{1})$ with
420 composition $|$ together with a function $M : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{C}$. Given such a structure, we define the
421 Honda & Yoshida's combinators of \mathbb{C} by means of:

$$\begin{array}{c}
\text{K}(a) \triangleq \lambda_{x \in \mathbb{N}} (M(a, x) \triangleright \mathbb{1}) \quad \left| \quad \text{Br}(a, b) \triangleq \lambda_{x \in \mathbb{N}} (M(a, x) \triangleright F(b, x)) \right. \\
\text{F}(a, b) \triangleq \lambda_{x \in \mathbb{N}} (M(a, x) \triangleright M(b, x)) \quad \left| \quad \text{D}(a, b, c) \triangleq \lambda_{x \in \mathbb{N}} (M(a, x) \triangleright M(b, x) | M(c, x)) \right. \\
\text{Bl}(a, b) \triangleq \lambda_{x \in \mathbb{N}} (M(a, x) \triangleright F(x, b)) \quad \left| \quad \text{S}(a, b, c) \triangleq \lambda_{x \in \mathbb{N}} (M(a, x) \triangleright F(b, c)) \right.
\end{array}$$

$$\begin{array}{c}
\frac{(x : P^\perp) \in \Gamma}{\vdash x : P \mid \Gamma} \text{(Ax}_+\text{)} \quad \frac{\vdash t : A \mid \Gamma \quad \vdash u : B \mid \Gamma}{\vdash (t, u) : A \otimes B \mid \Gamma} \text{(}\otimes\text{)} \quad \frac{\vdash t : A[P/X] \mid \Gamma}{\vdash t : \exists X.A \mid \Gamma} \text{(}\exists\text{)} \\
\frac{(\alpha : N^\perp) \in \Gamma}{\vdash \alpha : N \mid \Gamma} \text{(Ax}_-\text{)} \quad \frac{c : (\vdash \kappa : A, \kappa' : B, \Gamma)}{\vdash \mu(\kappa, \kappa').c : A \wp B \mid \Gamma} \text{(}\wp\text{)} \quad \frac{\vdash V : A \mid \Gamma \quad X \notin FV(\Gamma)}{\vdash V : \forall X.A \mid \Gamma} \text{(}\forall\text{)} \\
\frac{c : (\vdash \kappa : A, \Gamma)}{\vdash \mu\kappa.c : A \mid \Gamma} \text{(}\mu\text{)} \quad \frac{\vdash t : A \mid \Gamma \quad \vdash u : A^\perp \mid \Gamma}{\langle t \parallel u \rangle : (\vdash \Gamma)} \text{(Cut)}
\end{array}$$

■ **Figure 3** System L, typing rules

423 These definitions are sound w.r.t. the expected reductions of these combinators, in the
424 sense that each reduction rule $p \rightarrow q$ (in [9]) yields an inequality $p \preceq q$.

425 ► **Proposition 45.** *Given a HYS $(\mathbb{C}, \preceq, \otimes, |, (\cdot)^\perp, 1, \mathbf{M})$ we have:*

$$\begin{array}{c}
\begin{array}{l}
\text{K}(a)|\mathbf{M}(a, x) \preceq 1 \\
\text{F}(a, b)|\mathbf{M}(a, x) \preceq \mathbf{M}(b, x) \\
\text{D}(a, b, c)|\mathbf{M}(a, x) \preceq \mathbf{M}(b, x)|\mathbf{M}(c, x)
\end{array} \quad \left| \quad \begin{array}{l}
\text{Bl}(a, b)|\mathbf{M}(a, x) \preceq \text{F}(x, b) \\
\text{Br}(a, b)|\mathbf{M}(a, x) \preceq \text{F}(b, x) \\
\text{S}(a, b, c)|\mathbf{M}(a, x) \preceq \text{F}(b, c)
\end{array}
\end{array}$$

427 ► **Definition 46.** *A Honda & Yoshida's Algebra (HYA) is a HYS $(\mathbb{C}, \preceq, \otimes, |, (\cdot)^\perp, 1, \mathbf{M})$
428 together with a monoidal separator $\mathcal{S} \subseteq \mathbb{C}$ s.t. all Honda & Yoshida combinators belong to \mathcal{S} .*

429 So far we have not needed to introduce a notion of reduction because the constructors
430 of Linear Logic in our types reflect the properties of connections and handling of names.
431 However, to obtain a model of behaviors \mathcal{B} where the Honda & Yoshida's combinators are
432 inhabited, we ask the poles to be closed by anti-reduction. This choice follows the first
433 author's design when defining the *regular poles* in [1].

434 ► **Definition 47.** *Let us consider a pole $\perp \subseteq \bar{\Pi}$. We say that \perp is regular iff for all
435 $(P, e), (Q, f)$ (PGF)-processes and names u, \vec{x} , $\{(P \mid Q, ef)\}^\perp \subseteq \{(u^\uparrow \langle \vec{x} \rangle . P \mid v^\perp(\vec{x}). Q, ef)\}^\perp$
436 whenever $u \underset{ef}{\sim} v$.*

437 ► **Proposition 48.** *Let us consider a CIMA $(\mathbb{C}, \preceq, \otimes, ||, (\cdot)^\perp, 1, \mathcal{S})$ whose pole \perp is regular.
438 Then \mathbb{C} is a HYA with the definition $\mathbf{M}(a, b) \triangleq \{(a^\uparrow \langle b \rangle . \perp, \Delta_{\mathbb{N}})\}^{\perp\perp}$.*

439 **Proof.** It suffices to prove that all combinators of Definition 44 are inhabited. For that we
440 consider the standard encoding of the negative combinators into the π -calculus (which is
441 embedded into PGF). Then, using the regularity of the pole, we prove that the encoding
442 belongs to the corresponding combinator defined in \mathbb{C} (see on the appendix). ◀

443 6 Parallel composition cannot be derived

444 We shall now see that the extra structure presented in the previous section to encompass
445 parallelism within conjunctive involutive monoidal algebra is actually necessary, in the sense
446 that parallelism cannot be derived from the ground structure of a conjunctive involutive
447 monoidal algebra. We do so by providing a concrete example of a conjunctive involutive
448 monoidal algebra in which we prove that the axiom of Definition 40 are not satisfied.

449 We build on a realizability interpretation based on a fragment of Munch-Maccagnoni's
450 system L [16], a polarised sequent calculus tailored to give a term language to Girard's classical

451 logic LC. To draw the comparison with other sequent calculi such as Curien-Herbelin's $\bar{\lambda}\mu\tilde{\mu}$ -
 452 calculus [3], the main difference lies in the use of polarisation, which makes lazy and strict
 453 qualify logical connectives rather than evaluation strategies: instead of having to enforce
 454 globally a strategy to avoid critical pair, lazy and strict evaluations coexist and are dictated
 455 locally by the polarities of the corresponding connectives. For a more detailed introduction
 456 on the rationale of \mathbf{L} , we refer the interested reader to [17].

457 We work here with a fragment of \mathbf{L} , keeping only the connectives necessary to induce a
 458 CIMA. As explained above, formulas are divided into positive and negative ones:

$$459 \begin{array}{l} \text{Positive formulas} \\ \text{Negative formulas} \end{array} \begin{array}{l} P ::= X \mid A \otimes B \mid \exists X.A \\ N ::= X^\perp \mid A \wp B \mid \forall X.A \end{array} \quad \left| \quad \begin{array}{l} \text{Formulas} \\ A, B ::= P \mid N \end{array} \right.$$

460 To each positive formula P is associated a dual negative formula P^\perp (and vice-versa), defined
 461 by induction on the syntax of formulas. As is usual in linear logic, the map $(\cdot)^\perp$ defines an
 462 involutive function on formulas:

$$463 \begin{array}{l} (X)^\perp = X^\perp \\ (X^\perp)^\perp = X \end{array} \quad \left| \quad \begin{array}{l} (A \otimes B)^\perp = A^\perp \wp B^\perp \\ (A \wp B)^\perp = A^\perp \otimes B^\perp \end{array} \right. \quad \left| \quad \begin{array}{l} (\forall X.A)^\perp = \exists X.A^\perp \\ (\exists X.A)^\perp = \forall X.A^\perp \end{array} \right.$$

464 Similarly, the syntax of terms and values reflect these distinction between negative and
 465 positive connectives. To ease the connection with [16], we stick to the same presentation:

$$466 \begin{array}{l} \text{Variables} \\ \text{Terms} \end{array} \begin{array}{l} \kappa ::= x \mid \alpha \\ t_- ::= \alpha \mid \mu x.c \mid \mu(\kappa, \kappa).c \\ t_+ ::= x \mid \mu\alpha.c \mid (t, u) \\ t ::= t_+ \mid t_- \end{array} \quad \left| \quad \begin{array}{l} \text{Commands} \\ \text{Values} \end{array} \begin{array}{l} c ::= \langle t_+ \parallel t_- \rangle \mid \langle t_- \parallel t_+ \rangle \\ V_+ ::= x \mid (V, V') \\ V ::= V_+ \mid t_- \end{array} \right.$$

467 In the sequel, we write \mathcal{T}_+ (resp. $\mathcal{T}_-, \mathcal{C}, \mathcal{V}$) for the set of positive terms (resp. negative terms,
 468 commands, values), and denote by \mathcal{T}_+^0 , etc... the corresponding set of closed terms. As in [16],
 469 we use monolateres sequents, following Girard's tradition to have all the formulas on the right.
 470 To that end, we need to quotient the syntax with a new α -equivalence $\langle t \parallel u \rangle \equiv \langle u \parallel t \rangle$, which
 471 will simplify overall the presentation. The type system, which distinguishes between sequents
 472 of the shape $\vdash t : A \mid \Gamma$ to type terms (where A is the formula *in the stoup*) and sequents of
 473 the shape $c : (\vdash \Gamma)$ to type commands, is given in Figure 3. The reader may observe that
 474 the rule for the universal quantifier has to be restricted to values in order to avoid the usual
 475 inconsistency of polymorphism in presence of side-effects [16].

476 The operational semantics is defined as a weak-head reduction relation on commands, by
 477 means of the following reduction rules:

$$478 \begin{array}{l} \langle \mu\alpha.c \parallel V \rangle \rightarrow_\mu c[V/x] \\ \langle V \parallel \mu x.c \rangle \rightarrow_\mu c[V/x] \end{array} \quad \left| \quad \begin{array}{l} \langle (V, V') \parallel \mu(\kappa, \kappa').c \rangle \rightarrow_\beta c[V/\kappa, V'/\kappa'] \\ \langle (t, t') \parallel u \rangle \rightarrow_\xi \langle t \parallel \mu\kappa. \langle t' \parallel \mu\kappa'. \langle (\kappa, \kappa') \parallel u \rangle \rangle \rangle \end{array} \right.$$

479 The different binders only reduce in front of values, while the \rightarrow_ξ rules specify how constructors
 480 should be expanded when they are build on top of a term instead of a value. In particular, it
 481 is worth noting that the reduction $\rightarrow_\xi \triangleq \rightarrow_\mu \cup \rightarrow_\beta \cup \rightarrow_\xi$ has no critical pair, which entails the
 482 Church-Rosser property when considering its extension to subcommands.

483 Following [16], we can now define a realizability interpretation *à la* Krivine relying on this
 484 calculus. As is usual in Krivine realizability, we say that a *pole* is any subset of \mathcal{C}^0 that is
 485 closed by anti-reduction. This induces an orthogonality relation on terms, which we extend
 486 to any sets $T \in \mathcal{P}(\mathcal{T}_+^0)$ or $U \in \mathcal{P}(\mathcal{T}_-^0)$ by defining:

$$487 T^\perp \triangleq \{t_- \in \mathcal{T}_-^0 : \forall t_+ \in T, \langle t_+ \parallel t_- \rangle \in \perp\} \quad \left| \quad U^\perp \triangleq \{t_+ \in \mathcal{T}_+^0 : \forall t_- \in U, \langle t_+ \parallel t_- \rangle \in \perp\}$$

488 In this context, we call *behavior* any subset T of \mathcal{T}_+^0 or \mathcal{T}_-^0 such that $T = T^{\perp\perp}$ and we denote
489 by \mathbf{H} the set of all behaviors. Observe that $(\cdot)^\perp$ defines an involutive map on \mathbf{H} .

490 We are now ready to define a realizability interpretation that will associate to any formula
491 a behavior. More precisely, we extend the language of formulas to consider formulas with
492 (positive) parameters R in the set $\Pi \triangleq \mathcal{P}(\mathcal{T}_+^0 \cap \mathbb{V})$, and to any such positive formula P we
493 associate a behavior $|P| \in \mathcal{P}(\mathcal{T}_+^0)$ while to any negative formula N we associate a behavior
494 $|N| \in \mathcal{P}(\mathcal{T}_-^0)$. The interpretation $|\cdot|$ is defined by induction on formulas:

$$\begin{array}{l}
 |R| \triangleq R^{\perp\perp} \\
 |A \otimes B| \triangleq (|A| \otimes |B|)^{\perp\perp} \\
 |\exists X.A| \triangleq (\bigcup_{R \in \mathbb{V}} |A[R/X]|_{\mathbb{V}})^{\perp\perp}
 \end{array}
 \quad \left| \quad
 \begin{array}{l}
 |R^\perp| \triangleq R^\perp \\
 |A \wp B| \triangleq (|A^\perp| \otimes |B^\perp|)^\perp \\
 |\forall X.A| \triangleq (\bigcap_{R \in \mathbb{V}} |A[R/X]|_{\mathbb{V}})^{\perp\perp}
 \end{array}
 \right.$$

496 where $|A|_{\mathbb{V}} \triangleq |A| \cap \mathbb{V}$ and $T \otimes U \triangleq \{(t, u) : t \in T \wedge u \in U\}$. Observe that by construction,
497 for any closed formula A , it holds that $|A^\perp| = |A|^\perp$.

498 ► **Definition 49.** For any term t , one says that t realizes a formula A whenever $t \in |A|$,
499 which we denote by $t \Vdash A$. Similarly, we write $\sigma \Vdash \Gamma$ to denote that a substitution σ realizes
500 a context $\Gamma \equiv x_1 : A_1, \dots, x_n : A_n$ when for any $1 \leq i \leq n$, $\sigma(x_i) \Vdash A_i$.

501 We say that t is a universal realizer of A and we write $t \Vdash\!\!\! \Vdash A$ when a term t realizes A
502 for any possible choice of pole.

503 We will not develop any further on the properties of this construction, but we shall at
504 least emphasize that the typing rules defined in Figure 3 are adequate with respect to this
505 interpretation, which is proven as usual by induction over typing derivations [16].

506 ► **Proposition 50 (Adequacy).** Let Γ be a typing context, \perp a pole and σ be a substitution
507 such that $\sigma \Vdash \Gamma$, then for any term t and formula A , if $\vdash t : A \mid \Gamma$, then $t \Vdash A$.

508 Even though the fragment of \mathbf{L} we have selected does not account for linearity, the last
509 proposition shows that the realizability interpretation still defines a model for MLL. In
510 particular, as was observed in [15], algebraically it induces a conjunctive structure. This
511 observation can be adapted to our framework, to show that the realizability interpretation
512 defined above actually induces a conjunctive involutive monoidal algebra.

513 ► **Proposition 51.** The tuple $(\mathbf{H}, \subseteq, \otimes, (\cdot)^\perp, \perp, \mathcal{S})$, where $\perp \triangleq \mathcal{T}_+^0$ and $\mathcal{S} = \mathbf{H} \setminus \emptyset$, is a CIMA.

514 **Proof.** The proof is analogous to the proof in [15], up to the observation that $(\cdot)^\perp$ is involutive
515 on \mathbf{H} . In particular, the axioms to define a separator are realized by similar terms (including
516 the new ones S_6, S_7, S_8 which are valid in the internal logic of classical conjunctive algebras
517 in [15]). See Appendix G for a complete proof. ◀

518 We have shown so far that the realizability interpretation based on \mathbf{L} defines a conjunctive
519 involutive monoidal algebra as expected. Besides, we can show that this particular kind of
520 CIMA does not admit parallelism, thus justifying the necessity of the extra requirements
521 formulated in Definition 40. Indeed, to obtain a compositional structure in such a model
522 without adding some extra structure would mean the desired composition F could be defined
523 by means of a term t (i.e. $F(u, v) \triangleq t u b$) such that the expected axioms are satisfied.

524 We prove that this is not possible, in that the first two axioms ($F(a, \perp) \equiv a$ and
525 $F(\perp, a) \equiv a$) cannot be satisfied together. The proof, which is folklore in Krivine realizability,
526 mostly consists in proving that a term that realize both of these axioms should essentially
527 behave like a parallel \wp operator, which does not exist in \mathbf{L} . This should not come as a
528 surprise, since, to put it differently, it essentially means that to get the algebraic structure
529 corresponding to parallelism, we actually need to have parallel computations in our calculus.

530 ► **Proposition 52.** *There is no term $t \in \mathcal{T}_+^0$ s.t. $t \Vdash \forall X.(X \multimap \mathbb{1} \multimap X) \wedge (\mathbb{1} \multimap X \multimap X)$.*

531 **Proof.** To simplify the proof, let us assume that the calculus is extended with two inert
532 (positive) constants κ_1, κ_2 and a negative one α (alternatively, one could consider any terms
533 $u_1, u_2 \in \mathcal{T}_+^0$ and $v \in \mathcal{T}_-^0$ such that the commands $\langle u_1 \parallel v \rangle$ and $\langle u_2 \parallel v \rangle$ are blocked).

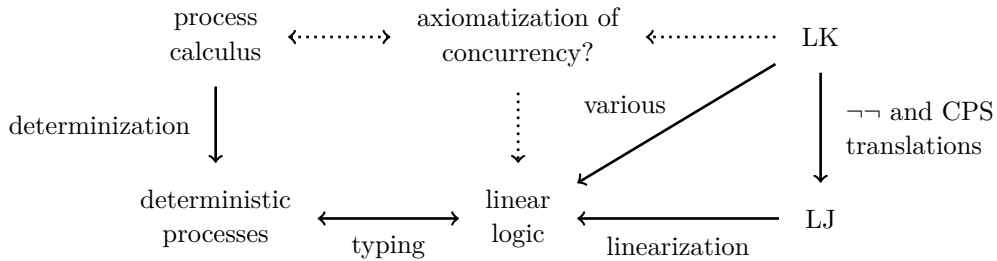
534 We reason by contradiction by assuming that such a term t exists. Let $u \in \mathcal{T}_-^0$ be any
535 term, and $\mathbb{X} \triangleq \{u\} \in \Pi$. Consider the pole $\perp_1 = \{c : c \rightarrow \langle \kappa_1 \parallel u \rangle\}$. By construction, we
536 have that $\kappa_1 \in |\mathbb{X}|$, $\kappa_2 \in |\perp_1|$ and $\alpha \in |\mathbb{X}|^\perp$. Since by assumption $t \Vdash \mathbb{X} \multimap \mathbb{1} \multimap \mathbb{X}$, we get
537 $\langle t \parallel (\kappa_1, (\kappa_2, \alpha)) \rangle \in \perp_1$, i.e. $\langle t \parallel (\kappa_1, (\kappa_2, \alpha)) \rangle \rightarrow \langle \kappa_1 \parallel \alpha \rangle$.

538 By taking $\perp_2 = \{c : c \rightarrow \langle \kappa_2 \parallel \alpha \rangle\}$, we prove similarly that $\langle t \parallel (\kappa_1, (\kappa_2, \alpha)) \rangle \rightarrow \langle \kappa_2 \parallel \alpha \rangle$.
539 Since both commands $\langle \kappa_1 \parallel \alpha \rangle$ and $\langle \kappa_2 \parallel \alpha \rangle$ are blocked and the calculus is determinist, we
540 indeed found a contradiction. ◀

541 7 Conclusion & future work

542 **Exponentials.** In addition to a study more in depth of the properties of the several struc-
543 tures introduced in this paper, a natural prolongation of this work would be to investigate
544 the possibility to build on conjunctive involutive monoidal algebra to account for more
545 expressive logical systems, in particular MALL with exponentials. This would provide us
546 with a complete algebraic presentation of the first author’s work [1]. To that end, we could
547 rely on Honda & Yoshida’s combinatorial approach to replication [10]. Another path to follow
548 in that direction would consist in exploring the connections with the Geometry of Interaction
549 as it is presented in Duchesne’s work [4], where MLL is interpreted using partial permutations
550 and the tensor corresponds to an appropriate disjoint union. This bears similarities with our
551 interpretation using injections $\iota_i : \mathbb{N} \rightarrow \mathbb{N}$ for the tensor to ensure disjoint namespaces.

552
553 **Concurrency.** A central element in our study of parallelism and its axiomatization is
554 the role of non-determinism. Indeed, the presence in a process of several actions that may
555 synchronize together in different ways is both a source of expressiveness, as witnessed by the
556 famous “parallel or” which relies on this feature, but also an obstacle for logical study. In
557 some sense, our search for an axiomatization of parallel composition aims at identifying the
558 part of the logical structure that is responsible for such phenomena. An analogy can be made
559 with the computational interpretation of classical logic: it is well known that the classical
560 sequent calculus LK is not confluent, to the point that any direct denotational semantics of
561 proofs is degenerated, but interesting semantics can be obtained by means of translations
562 into better behaved logics. Such translations are effective because they impose constraints
563 on proof reduction, like call-by-name or call-by-value policies in languages with control. Our
564 system with parallelism could be envisaged as a linear analogue of full classical logic with no
565 constraint on strategies.



566

567 — References

- 568 1 Emmanuel Beffara. *Logique, réalisabilité et concurrence*. PhD Thesis, Université Paris 7,
569 December 2005.
- 570 2 G. Birkhoff. *Lattice theory*, volume 25 of *Colloquium publications*. American Mathematical
571 Society, 1940.
- 572 3 Pierre-Louis Curien and Hugo Herbelin. The duality of computation. In *Proceedings of ICFP*
573 *2000*, SIGPLAN Notices 35(9), pages 233–243. ACM, 2000. doi:10.1145/351240.351262.
- 574 4 Etienne Duchesne. *La localisation en logique : géométrie de l'interaction et sémantique dénota-*
575 *tionnelle*. PhD thesis, 2009. Thèse de doctorat dirigée par Régnier, Laurent Mathématiques
576 Aix-Marseille 2 2009. URL: <http://www.theses.fr/2009AIX22080>.
- 577 5 Walter Ferrer Santos, Jonas Frey, Mauricio Guillermo, Octavio Malherbe, and Alexandre
578 Miquel. Ordered combinatory algebras and realizability. *Mathematical Structures in Computer*
579 *Science*, 27(3):428–458, 2017. doi:10.1017/S0960129515000432.
- 580 6 Yuxi Fu. The χ -Calculus. In *1997 Advances in Parallel and Distributed Computing Conference*
581 *(APDC)*, pages 74–81. Computer Society Press, 1997.
- 582 7 Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1):1 – 101, 1987. doi:
583 10.1016/0304-3975(87)90045-4.
- 584 8 Jean-Yves Girard. Locus Solum: From the rules of logic to the logic of rules. *Mathematical*
585 *Structures in Computer Science*, 11(3):301–506, June 2001. doi:10.1017/S096012950100336X.
- 586 9 Kohei Honda and Nobuko Yoshida. Combinatory representation of mobile processes. In
587 *ACM-SIGACT Symposium on Principles of Programming Languages*, 1994.
- 588 10 Kohei Honda and Nobuko Yoshida. Replication in concurrent combinators. In Masami Hagiya
589 and John C. Mitchell, editors, *Theoretical Aspects of Computer Software*, pages 786–805,
590 Berlin, Heidelberg, 1994. Springer Berlin Heidelberg.
- 591 11 J.M.E. Hyland. The effective topos. *Studies in Logic and the Foundations of Mathem-*
592 *atics*, 110:165 – 216, 1982. The L. E. J. Brouwer Centenary Symposium. doi:10.1016/
593 S0049-237X(09)70129-6.
- 594 12 J.-L. Krivine. Realizability in classical logic. In *Interactive models of computation and program*
595 *behaviour. Panoramas et synthèses*, 27:197–229, 2009. Publisher: Société Mathématique de
596 France.
- 597 13 Jean-Louis Krivine. Realizability algebras II : new models of ZF + DC. *Logical Methods in*
598 *Computer Science*, 8(1):10, February 2012. 28 p. doi:10.2168/LMCS-8(1:10)2012.
- 599 14 Alexandre Miquel. Implicative algebras: a new foundation for realizability and forcing. *Mathem-*
600 *atical Structures in Computer Science*, 30(5):458–510, 2020. doi:10.1017/S0960129520000079.
- 601 15 Étienne Miquey. Revisiting the Duality of Computation: An Algebraic Analysis of Clas-
602 sical Realizability Models. In Maribel Fernández and Anca Muscholl, editors, *28th EACSL*
603 *Annual Conference on Computer Science Logic (CSL 2020)*, volume 152 of *Leibniz Inter-*
604 *national Proceedings in Informatics (LIPIcs)*, pages 30:1–30:18, Dagstuhl, Germany, 2020.
605 Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. URL: [https://drops.dagstuhl.de/opus/](https://drops.dagstuhl.de/opus/volltexte/2020/11673)
606 [volltexte/2020/11673](https://drops.dagstuhl.de/opus/volltexte/2020/11673), doi:10.4230/LIPIcs.CSL.2020.30.
- 607 16 Guillaume Munch-Maccagnoni. Focalisation and Classical Realisability. In Erich Grädel and Re-
608 inhard Kahle, editors, *Computer Science Logic '09*, volume 5771 of *Lecture Notes in Computer*
609 *Science*, pages 409–423. Springer, Heidelberg, 2009. doi:10.1007/978-3-642-04027-6_30.
- 610 17 Guillaume Munch-Maccagnoni. *Syntax and Models of a non-Associative Composition of*
611 *Programs and Proofs*. PhD thesis, Univ. Paris Diderot, 2013.
- 612 18 Joachim Parrow and Björn Victor. The Fusion Calculus: Expressiveness and Symmetry in
613 Mobile Processes. In *13th IEEE Symposium on Logic in Computer Science (LICS)*, pages
614 176–185, 1998.
- 615 19 Andrew M. Pitts. Tripos theory in retrospect. *Mathematical Structures in Computer Science*,
616 12(3):265–279, 2002. doi:10.1017/S096012950200364X.
- 617 20 Davide Sangiorgi and David Walker. *The π -Calculus: A Theory of Mobile Processes*. Cambridge
618 University Press, 2001.

- 619 21 Thomas Streicher. Krivine's classical realisability from a categorical perspective. *Mathematical*
620 *Structures in Computer Science*, 23(6):1234–1256, 2013. doi:10.1017/S0960129512000989.
- 621 22 Lucian Wischik and Philippa Gardner. Explicit fusions. *Theoretical Computer Science*,
622 340(3):606–630, August 2005. doi:10.1016/j.tcs.2005.03.017.

A

 Logic in conjunctive structures

We give here the proofs of Section 2, with all the details, but the proofs are mostly simple combinatorial manipulations within CISs.

► **Definition 53.** *Let us consider a conjunctive involutive structure. We define:*

$$\begin{aligned} S'_3 &\triangleq \lambda_{a,b \in \mathbb{C}} (a \wp b) \multimap (b \wp a) \\ S'_4 &\triangleq \lambda_{a,b,c \in \mathbb{C}} (a \multimap b) \multimap (b \multimap c) \multimap a \multimap c \\ t \circ s &\triangleq S'_4 * s * t \end{aligned}$$

It is easy to check that:

► **Lemma 54.** *For any CIMA $(\mathbb{C}, \mathcal{S})$, we have $S'_3, S'_4 \in \mathcal{S}$ and \mathcal{S} is closed under \circ .*

Proof. As for implicative algebras, separators of CIMAs are closed under application $*$, therefore they contains S'_3, S'_4 and are also closed under composition \circ . ◀

► **Proposition 55.** *Let us consider a CIMA $(\mathbb{C}, \mathcal{S})$. The following are members of \mathcal{S} :*

1. $I \triangleq \lambda_{a \in \mathbb{C}} a \multimap a$.
2. $\lambda_{a,b,c \in \mathbb{C}} (a \otimes (b \otimes c)) \multimap ((a \otimes b) \otimes c)$.
3. $\lambda_{a,b \in \mathbb{C}} a \multimap b \multimap (a \otimes b)$.
4. $\lambda_{a,b \in \mathbb{C}} (a \wp b) \multimap (b \wp a)$.
5. $\lambda_{a,b \in \mathbb{C}} (a \multimap b) \multimap (a \wp c) \multimap (b \wp c)$.
6. $\lambda_{a,b,c \in \mathbb{C}} (a \wp b) \wp c \multimap a \wp (b \wp c)$ and $\lambda a \wp (b \wp c) \multimap (a \wp b) \wp c$.

Proof.

1. For any $a \in \mathbb{C}$ we have $S_6 \preceq a \multimap a \otimes \mathbf{1}$ and $S_7 \preceq a \otimes \mathbf{1} \multimap a$. Then $S_7 \circ S_6 \preceq a \multimap a$ (uniformly on $a \in \mathbb{C}$) and by Lemma 54 we get $S_7 \circ S_6 \in \mathcal{S}$ and then $\lambda_{a \in \mathbb{C}} a \multimap a \in \mathcal{S}$.
2. For all $a, b, c \in \mathbb{C}$ we have:

$$\begin{aligned} S_3 &\preceq a \otimes (b \otimes c) \multimap (b \otimes c) \otimes a \\ S_5 &\preceq (b \otimes c) \otimes a \multimap b \otimes (c \otimes a) \\ S_3 &\preceq b \otimes (c \otimes a) \multimap (c \otimes a) \otimes b \\ S_5 &\preceq (c \otimes a) \otimes b \multimap c \otimes (a \otimes b) \\ S_3 &\preceq c \otimes (a \otimes b) \multimap (a \otimes b) \otimes c \end{aligned}$$

Then $S_3 \circ S_5 \circ S_3 \circ S_5 \circ S_3 \preceq a \otimes (b \otimes c) \multimap (a \otimes b) \otimes c$ for all $a, b, c \in \mathbb{C}$ and by Lemma 54 we get $S_3 \circ S_5 \circ S_3 \circ S_5 \circ S_3 \in \mathcal{S}$, thus concluding $\lambda_{a,b,c \in \mathbb{C}} a \otimes (b \otimes c) \multimap (a \otimes b) \otimes c \in \mathcal{S}$.

3. By definition $a \multimap b \multimap (a \otimes b) = \left(a \otimes (b \otimes (a \otimes b)^\perp) \right)^\perp$. It suffices to find $U \in \mathcal{S}$ uniform on a, b s.t. $U \preceq \left(a \otimes (b \otimes (a \otimes b)^\perp) \right)^\perp$. By Item 2, $T \preceq a \otimes (b \otimes (a \otimes b)^\perp) \multimap (a \otimes b) \otimes (a \otimes b)^\perp$ for some $T \in \mathcal{S}$. By monotonicity of $*$, $S_8 * T \preceq \left((a \otimes b) \otimes (a \otimes b)^\perp \right)^\perp \multimap \left(a \otimes (b \otimes (a \otimes b)^\perp) \right)^\perp$ and then $S_8 * T * I \preceq \left(a \otimes (b \otimes (a \otimes b)^\perp) \right)^\perp = a \multimap b \multimap (a \otimes b)$. By Lemma 54 we get the result.

4. $S_8 * S_3 \preceq \lambda_{a,b \in \mathbb{C}} (a \wp b) \multimap (b \wp a)$ and by Lemma 54 we get the result.

5. On one hand $(a \multimap b) \multimap (a \wp c) \multimap (b \wp c) = (a \multimap b) \multimap (a^\perp \otimes c^\perp)^\perp \multimap (b^\perp \otimes c^\perp)^\perp$. On the other hand we have:

$$\begin{aligned} S_3 &\preceq (b^\perp \multimap a^\perp) \multimap (b^\perp \otimes c^\perp) \multimap (a^\perp \otimes c^\perp) \\ S_3 \circ S_8 &\preceq (a \multimap b) \multimap (b^\perp \otimes c^\perp) \multimap (a^\perp \otimes c^\perp) \\ S_8 \circ S_3 \circ S_8 &\preceq (a \multimap b) \multimap (a^\perp \otimes c^\perp)^\perp \multimap (b^\perp \otimes c^\perp)^\perp \end{aligned}$$

By Lemma 54 we get the result.

656 6. On one hand $(a \wp b) \wp c \multimap a \wp (b \wp c) = \left((a^\perp \otimes b^\perp) \otimes c^\perp \right)^\perp \multimap \left(a^\perp \otimes (b^\perp \otimes c^\perp) \right)^\perp$.
 657 On the other hand $T \preceq a^\perp \otimes (b^\perp \otimes c^\perp) \multimap (a^\perp \otimes b^\perp) \otimes c^\perp$ for some $T \in \mathcal{S}$ and
 658 $S_8 * T \preceq \left((a^\perp \otimes b^\perp) \otimes c^\perp \right)^\perp \multimap \left(a^\perp \otimes (b^\perp \otimes c^\perp) \right)^\perp$. By Lemma 54 we get the result.

659

660 ► **Proposition 56.** For any CIMA $(\mathbb{C}, \mathcal{S})$, $\lambda_{a,b,c \in \mathbb{C}} (a \wp b) \otimes c \multimap a \wp (b \otimes c) \in \mathcal{S}$

Proof. We must prove $\bigwedge_{a,b,c \in \mathbb{C}} \left(\left((a^\perp \otimes b^\perp)^\perp \otimes c \right) \otimes (a^\perp \otimes (b \otimes c)^\perp) \right)^\perp \in \mathcal{S}$. On one hand we
 have $S_5 \preceq \left((a^\perp \otimes b^\perp)^\perp \otimes c \right) \otimes (a^\perp \otimes (b \otimes c)^\perp) \multimap (a^\perp \otimes b^\perp)^\perp \otimes (c \otimes (a^\perp \otimes (b \otimes c)^\perp))$. By
 Proposition 55.2 $T_1 \preceq (a^\perp \otimes b^\perp)^\perp \otimes (c \otimes (a^\perp \otimes (b \otimes c)^\perp)) \multimap (a^\perp \otimes b^\perp)^\perp \otimes ((c \otimes a^\perp) \otimes (b \otimes c)^\perp)$ for
 some $T_1 \in \mathcal{S}$. Using the combinators S_3, S_4 we have $T_2 \preceq (a^\perp \otimes b^\perp)^\perp \otimes ((c \otimes a^\perp) \otimes (b \otimes c)^\perp) \multimap$
 $(a^\perp \otimes b^\perp)^\perp \otimes ((b \otimes c)^\perp \otimes (a^\perp \otimes c))$ for some $T_2 \in \mathcal{S}$. By Proposition 55.2 using the combinators
 S_3, S_4 we have $T_3 \preceq \left((a^\perp \otimes b^\perp)^\perp \otimes c \right) \otimes (a^\perp \otimes (b \otimes c)^\perp) \multimap (a^\perp \otimes b^\perp)^\perp \otimes ((b \otimes c)^\perp \otimes (a^\perp \otimes c))$
 for some $T_3 \in \mathcal{S}$ and $T_4 \preceq \left((a^\perp \otimes b^\perp)^\perp \otimes ((b \otimes c)^\perp \otimes (a^\perp \otimes c)) \right)^\perp \multimap \left((a^\perp \otimes b^\perp)^\perp \otimes c \right) \otimes$
 $(a^\perp \otimes (b \otimes c)^\perp)^\perp$ for some $T_4 \in \mathcal{S}$. In other terms we have:

$$T_4 \preceq \left((a^\perp \multimap b) \multimap ((b \multimap c^\perp) \multimap (a^\perp \multimap c^\perp)) \right) \multimap \left((a \wp b) \otimes c \multimap a \wp (b \otimes c) \right)$$

661 From which we deduce $T_4 * S'_4 \preceq (a \wp b) \otimes c \multimap a \wp (b \otimes c)$ for all $a, b, c \in \mathbb{C}$ and hence
 662 $T_4 * S'_4 \preceq \lambda_{a,b,c \in \mathbb{C}} (a \wp b) \otimes c \multimap a \wp (b \otimes c)$, thus proving the result. ◀

663 ► **Proposition 57.** For any CIMA $(\mathbb{C}, \mathcal{S})$ and any permutation σ , we have:

- 664 1. $\mathbf{ex}(\sigma) \triangleq \lambda_{a_1, \dots, a_k \in \mathbb{C}} (a_1 \wp \dots \wp a_k) \multimap (a_{\sigma(1)} \wp \dots \wp a_{\sigma(k)}) \in \mathcal{S}$.
- 665 2. $I \in \mathcal{S}$.
- 666 3. $\mathbf{t} \triangleq \lambda_{a,b,g,d \in \mathbb{C}} (g \wp a) \multimap (b \wp d) \multimap g \wp ((a \otimes b) \wp d) \in \mathcal{S}$
- 667 4. $\mathbf{c} \triangleq \lambda_{a,g,d \in \mathbb{C}} (g \wp a) \multimap (a^\perp \wp d) \multimap (g \wp d) \in \mathcal{S}$.
- 668 5. $\mathbf{1} \in \mathcal{S}$.

669 **Proof.**

- 670 1. Is a consequence of $S'_3, S'_4 \in \mathcal{S}$, the Proposition 55.Item 6 and Lemma 54. For instance,
 671 let us prove that $\lambda_{a_1, a_2, a_3} a_1 \wp a_2 \wp a_3 \multimap a_3 \wp a_2 \wp a_1 \in \mathcal{S}$. By Proposition 55. 6
 672 $T_1 \preceq (a_1 \wp (a_2 \wp a_3)) \multimap ((a_2 \wp a_3) \wp a_1)$ for some $T_1 \in \mathcal{S}$. By Proposition 55.5 and
 673 Definition 8, $T_2 \preceq ((a_2 \wp a_3) \wp a_1) \multimap ((a_3 \wp a_2) \wp a_1)$ for some $T_2 \in \mathcal{S}$. By Proposition 55.6,
 674 $T_3 \preceq ((a_3 \wp a_2) \wp a_1) \multimap (a_3 \wp (a_2 \wp a_1))$ for some $T_3 \in \mathcal{S}$. By Lemma 54 $T_3 \circ T_2 \circ T_1 \in \mathcal{S}$
 675 and $T_3 \circ T_2 \circ T_1 \preceq (a_1 \wp a_2 \wp a_3) \multimap (a_3 \wp a_2 \wp a_1)$ for all $a_1, a_2, a_3 \in \mathbb{C}$, all which ends
 676 the proof.
- 677 2. By Proposition 55.1.
- 678 3. By Proposition 55.3, $T_0 \preceq (g \wp a) \multimap (b \wp d) \multimap (g \wp a) \otimes (b \wp d)$ for some $T_0 \in \mathcal{S}$. By
 679 Proposition 56, $T_1 \preceq (g \wp a) \otimes (b \wp d) \multimap g \wp (a \otimes (b \wp d))$ for some $T_1 \in \mathcal{S}$. By Definition 8,
 680 Proposition 55.5 and Proposition 56, $T_2 \preceq g \wp (a \otimes (b \wp d)) \multimap g \wp ((a \otimes b) \wp d)$ for some
 681 $T_2 \in \mathcal{S}$. By Lemma 54, $T_2 \circ T_1 \circ T_0 \in \mathcal{S}$ and $T_2 \circ T_1 \circ T_0 \preceq (g \wp a) \multimap (b \wp d) \multimap g \wp ((a \otimes b) \wp d)$
 682 for all $g, d, a, b \in \mathbb{C}$, all which ends the proof.
4. On one hand $S'_4 \preceq (g^\perp \multimap a) \multimap (a \multimap d) \multimap (g^\perp \multimap d)$ which in turn gives

$$S'_4 \preceq (g \wp a) \multimap (a^\perp \wp d) \multimap (g \wp d) \in \mathcal{S}$$

683 for all $g, d, a \in \mathbb{C}$ and hence $S'_4 \preceq \lambda_{g,d,a \in \mathbb{C}} (g \wp a) \multimap (a^\perp \wp d) \multimap (g \wp d) \in \mathcal{S}$.

684 5. By Definition 8, (unit)-rule. ◀

685 As a consequence, we directly deduce from the previous proposition the soundness of MLL
686 semantic typing rules:

687 ▶ **Theorem 10.** *The semantic typing rules of Figure 1 are sound for all CIMA.*

688 ▶ **Proposition 58.** *We pose $\text{comp} \triangleq \lambda_{a,b,c \in \mathbb{C}}(a \multimap b) \multimap (b \multimap c) \multimap a \multimap c$ and $t \circ u \triangleq$
689 $\text{comp}@t@u$. We have that $\text{comp} \in \mathcal{S}$ and if $t \preceq a \multimap b$ and $u \preceq b \multimap c$ implies that
690 $t \circ u \preceq a \multimap c$ (for any $t, u, a, b, c \in \mathbb{C}$).*

691 **Proof.** ◀

692 ▶ **Proposition 59.** 1. *For any $a, b \in \mathbb{C}$, if $a \in \mathcal{S}$ and $b \in \mathcal{S}$, then $a \otimes b \in \mathcal{S}$.*

693 ▶ **Proposition 38.** *The following behaviors are realized by pure fusions:*

- | | |
|---|--|
| 694 1. $\bigcap_{A \in \mathbb{B}} A \multimap A$.
2. $\bigcap_{A, B \in \mathbb{B}} (A \otimes B) \multimap (B \otimes A)$.
3. $\bigcap_{A, B, C \in \mathbb{B}} (A \multimap B) \multimap (B \multimap C) \multimap A \multimap C$.
4. $\bigcap_{A, B, C \in \mathbb{B}} ((A \otimes B) \otimes C) \multimap (A \otimes (B \otimes C))$. | 5. $\bigcap_{A \in \mathbb{B}} A \multimap (1 \otimes A)$.
6. $\bigcap_{A \in \mathbb{B}} (1 \otimes A) \multimap A$.
7. $\bigcap_{A, B \in \mathbb{B}} (A \multimap B) \multimap (B^\perp \multimap A^\perp)$. |
|---|--|

695 As we have seen, the first statement is the Lemma 36. We will prove the remaining
696 statements one by one in the following lemmas.

697 ▶ **Lemma 60.** *Let us consider $A, B \in \mathcal{P}(\bar{\Pi})$. Let us define the substitution $\tau : \mathbb{N}^1 \rightarrow \mathbb{N}^2$ s.t.
698 $\tau(n.1.1) \triangleq n.2.2$ and $\tau(n.2.1) \triangleq n.1.2$. Then $S_3' \triangleq \varepsilon_\tau \in (A \otimes B) \multimap (B \otimes A)$*

699 **Proof.** By Definition 32 $(A \otimes B) \multimap (B \otimes A) = \left((A \bullet B) \bullet (B \bullet A)^\perp \right)^\perp$. Consider $p \in (B \bullet A)^\perp$,
700 $q \in A$ and $r \in B$. By hypothesis we get $\bar{\nu}(r^1 | q^2 | p) \in \perp$. By definition $q^{1.1} | r^{2.1} | p^2 \in (A \otimes B) \multimap$
701 $(B \otimes A)$. It is enough to prove that $\varepsilon_\tau \perp q^{1.1} | r^{2.1} | p^2$, i.e.: $\bar{\nu}(q^{1.1} | r^{2.1} | p^2 | \varepsilon_\tau) \in \perp$. Moreover
702 $\bar{\nu}(q^{1.1} | r^{2.1} | p^2 | \varepsilon_\tau) \equiv (\nu \mathbb{N}^2)(\nu \mathbb{N}^1)(q^{1.1} | r^{2.1} | p^2 | \varepsilon_\tau) \equiv (\nu \mathbb{N}^2)(q^{2.2} | r^{1.2} | p^2) \equiv \bar{\nu}(r^1 | q^2 | p) \in \perp$,
703 which ends the proof. ◀

B Processes with global fusions

705 The main technical issue that needs to be addressed is how to compose fusions whose domains
706 lies on disjoint spaces of names with substitutions that link those spaces. The following
707 lemma is a fundamental result in this sense:

708 ▶ **Lemma 61.** *Let $X \subset \mathbb{N}$, $\tau : X \rightarrow X^c$ and $e, f \in \mathcal{E}$ with $|e| \subseteq X$ and $|f| \subseteq X^c$. Then:*

- 709 1. $ef\varepsilon_\tau \setminus X = e^\tau f$.
710 2. If $x \in X^c$ then $[x]_{ef\varepsilon_\tau} \setminus X = [x]_{e^\tau f}$. If $x \in X$ then $[x]_{ef\varepsilon_\tau} \setminus X = [\tau(x)]_{e^\tau f}$.

711 **Proof.** 1. We start by a remark: Let us consider $a \underset{ef\varepsilon_\tau}{\sim} b$ with $a, b \in X^c$. By definition of

712 $ef\varepsilon_\tau = \text{tcl} \left(e \cup f \cup \{(x, \tau(x)) \mid x \in X\} \right)$, there exists sequences S_0, \dots, S_k s.t. each S_j
713 has the shape $x_{4j} \underset{f}{\sim} x_{4j+1} \underset{\varepsilon_\tau}{\sim} x_{4j+2} \underset{e}{\sim} x_{4j+3} \underset{\varepsilon_\tau}{\sim} x_{4(j+1)}$ with $x_{4j}, x_{4j+1}, x_{4(j+1)} \in X^c$
714 and $x_{4j+2}, x_{4j+3} \in X$ and $a = x_0, x_{4(k+1)} \underset{f}{\sim} x_{4(k+1)+1} = b$:

$$\begin{array}{c}
X \\
\begin{array}{ccccc}
& & x_{4j+2} & \text{---} e & \text{---} & x_{4j+3} \\
& & \downarrow \varepsilon_\tau & & & \downarrow \varepsilon_\tau \\
X^c & x_{4j} & \text{---} f & \text{---} & x_{4j+1} & \text{---} e^\tau & \text{---} & x_{4(j+1)}
\end{array}
\end{array}$$

On the chain above observe that the sequences $x_{4j+1} \underset{\varepsilon_\tau}{\sim} x_{4j+2} \underset{e}{\sim} x_{4j+3} \underset{\varepsilon_\tau}{\sim} x_{4(j+1)}$ can be replaced by $x_{4j+1} \underset{\varepsilon_\tau}{\sim} x_{4(j+1)}$. This observation yields $ef\varepsilon_\tau \setminus X \subseteq e^\tau f$. The converse inclusion is straightforward since $e^\tau \subseteq e\varepsilon_\tau \setminus X$.

2. Let us consider $x \in X^c$. Given $y \in X^c$ by 1 it happens that $x \underset{ef\varepsilon_\tau}{\sim} y$ iff $x \underset{e^\tau f}{\sim} y$, thus concluding that $[x]_{ef\varepsilon_\tau \setminus X} = [x]_{e^\tau f}$. Let us consider $x \in X$. By definition of $ef\varepsilon_\tau$ we have $[x]_{ef\varepsilon_\tau} = [\tau(x)]_{ef\varepsilon_\tau}$ and then $[x]_{ef\varepsilon_\tau \setminus X} = [\tau(x)]_{ef\varepsilon_\tau \setminus X} = [\tau(x)]_{e^\tau f}$, the last equality by the previous part. \blacktriangleleft

The following is a technical requirement to prove Proposition 63

► **Lemma 62.** Consider $e, f \in \mathcal{E}$ with $|e| \subseteq X$ and $|f| \subseteq X^c$ for some set $X \subset \mathbb{N}$ and $\tau : X \rightarrow X^c$ a substitution. Clearly $[x]_{ef\varepsilon_\tau} \cap X^c \neq \emptyset$ for all $x \in \mathbb{N}$ and then $\mu_{(X, ef\varepsilon_\tau)}(x) = \min([x]_{ef\varepsilon_\tau \setminus X})$ for all $x \in \mathbb{N}$. By Lemma 61.2 we have $[x]_{ef\varepsilon_\tau \setminus X} = [\tau(x)]_{e^\tau f}$ if $x \in X$ and $[x]_{ef\varepsilon_\tau \setminus X} = [x]_{e^\tau f}$ if not. We conclude that

$$\mu_{(X, ef\varepsilon_\tau)}(x) = \begin{cases} (\tau(x))_{e^\tau f} & \text{if } x \in X \\ x_{e^\tau f} & \text{if } x \in X^c \end{cases}$$

► **Proposition 63.** Consider a substitution $\tau : X \rightarrow X^c$ for a set $X \subset \mathbb{N}$ and $p, q \in \bar{\Pi}$ such that $\text{FN}(p) \subseteq X$ and $\text{FN}(q) \subseteq X^c$. Then $(\nu X)(p \mid q \mid \varepsilon_\tau) \equiv_\alpha p^\tau \mid q$.

Proof. Observe that ε_τ makes that $[x]_{ef\varepsilon_\tau} \cap X^c \neq \emptyset$ for all $x \in \mathbb{N}$. Observe also that by Lemma 61.1 $ef\varepsilon_\tau \setminus X = e^\tau f$. Suppose that $p = (P, e)$ and $q := (Q, f)$. Applying Definition 26 we get $(\nu X)(p \mid q \mid \varepsilon_\tau) = (\nu X)(P \mid Q, ef\varepsilon_\tau) \equiv_\alpha (P^{\mu_{(X, ef\varepsilon_\tau)}} \mid Q^{\mu_{(X, ef\varepsilon_\tau)}}, e^\tau f)$ (the vector of names $x_{1, ef\varepsilon_\tau}^*, \dots, x_{h, ef\varepsilon_\tau}^*$ is empty). By Lemma 62 and the hypothesis $\text{FN}(P) \subseteq X$ and $\text{FN}(Q) \subseteq X^c$ we get $(P^{\mu_{(X, ef\varepsilon_\tau)}} \mid Q^{\mu_{(X, ef\varepsilon_\tau)}}, e^\tau f) \equiv_\alpha (P^\tau \mid Q, e^\tau f) = p^\tau \mid q$. \blacktriangleleft

The following is a technical prerequisite to prove Lemma 23:

► **Lemma 64.** Let us consider $x, y \in \mathbb{N}$, $e \in \mathcal{E}$ and $\sigma \triangleq \{y := y_e^*\}$, $\tau \triangleq \{x := x_{e \setminus \{y\}}^*\}$, $\sigma' \triangleq \{x := x_e^*\}$ and $\tau' \triangleq \{y := y_{e \setminus \{x\}}^*\}$. Then:

1. If $[x]_e = [y]_e = \{x, y\}$, then $\tau \circ \sigma = \{y := x\}$ and $\tau' \circ \sigma' = \{x := y\}$ and then they are equivalent by the transposition (xy) .
2. If $x \not\sim_e y$ or $[x]_e = [y]_e \supset \{x, y\}$, then $\tau \circ \sigma = \tau' \circ \sigma'$.

In both cases we conclude that $\tau \circ \sigma \sim \tau' \circ \sigma'$.

Proof. For $x = y$ is clear. If $x \neq y$ there are two cases:

1. $x \sim_e y$. This case splits into two subcases:

- a. $[x]_e = [y]_e = \{x, y\}$. Observe that $x_e^* = y$, $y_{e \setminus \{x\}}^* = y$, $y_e^* = x$ and $x_{e \setminus \{y\}}^* = x$ and then $\tau \circ \sigma = \sigma = \{y := x\}$ and $\tau' \circ \sigma' = \sigma' = \{x := y\}$ which are equivalent by means of the permutation (xy) (in fact a transposition).

749 b. $[x]_e = [y]_e \supset \{x, y\}$. Let us consider $z = \min([x]_e \setminus \{x, y\})$. Without loss of generality
750 we can suppose that $x < y$. Then we have three subcases:

751 i. $z < x < y$. In this case $x_e^* = z$, $y_{e \setminus \{x\}}^* = z$ and $y_e^* = z$, $x_{e \setminus \{y\}}^* = z$. Thus
752 $\sigma = \tau' = \{y := z\}$, $\sigma' = \tau = \{x := z\}$ and hence $\tau \circ \sigma = \tau' \circ \sigma'$ (which are equivalent
753 by means of the identity).

754 ii. $x < z < y$. In this case $x_e^* = z$, $y_{e \setminus \{x\}}^* = z$ and $y_e^* = x$, $x_{e \setminus \{y\}}^* = z$. Thus
755 $\sigma = \{y := x\}$, $\tau = \{x := z\}$, $\sigma' = \{x := z\}$ and $\tau' = \{y := z\}$ from which we get
756 $\tau \circ \sigma = \{y := z, x := z\} = \tau' \circ \sigma'$.

757 iii. $x < y < z$. In this case $x_e^* = y$, $y_{e \setminus \{x\}}^* = z$ and $y_e^* = x$, $x_{e \setminus \{y\}}^* = z$. Thus
758 $\sigma = \{y := x\}$, $\tau = \{x := z\}$, $\sigma' = \{x := y\}$ and $\tau' = \{y := z\}$ from which we get
759 $\tau \circ \sigma = \{y := z, x := z\} = \tau' \circ \sigma'$.

760 2. Suppose that $x \not\prec_e y$. Then we have that $x_e^* = x_{e \setminus \{y\}}^* = x'$ and $y_e^* = y_{e \setminus \{x\}}^* = y'$ for
761 some $x' \not\prec_e y'$. Thus $\sigma = \tau' = \{y := y'\}$, $\sigma' = \tau = \{x := x'\}$ from which we get
762 $\tau \circ \sigma = \{x := x', y := y'\} = \tau' \circ \sigma'$.

763

764 ► **Lemma 23.** Consider $(P, e) \in \bar{\Pi}$ and $x, y \in \mathbb{N}$, then $(\nu x)(\nu y)(P, e) \equiv_\alpha (\nu y)(\nu x)(P, e)$.

765 **Proof.** On one side:

$$766 (\nu x)(\nu y)(P, e) = (\nu x)((\nu y)P\{y := y_e^*\}, e \setminus \{y\}) = ((\nu xy)P\{y := y_e^*\}\{x := x_{e \setminus \{y\}}^*\}, e \setminus \{x, y\})$$

767 On the other hand, by Lemma 64, the last process is α -equivalent to

$$768 ((\nu yx)P\{x := x_e^*\}\{y := y_{e \setminus \{x\}}^*\}, e \setminus \{x, y\}) = (\nu y)((\nu x)P\{x := x_e^*\}, e \setminus \{x\}) = (\nu y)(\nu x)(P, e)$$

769 The generalization to finite sets is direct. ◀

770 C Concurrent realizability with PGF

771 We give here the proof of Section 4.1.

772 ► **Proposition 28.** For all $A, B \in \mathbb{P}$ and for any non-empty $\mathfrak{B} \subseteq \mathbb{P}$ we have:

773 1. If $A \subseteq B$ then $B^\perp \subseteq A^\perp$. 2. $A \subseteq A^{\perp\perp}$ and $A^\perp = A^{\perp\perp\perp}$. 3. $(\bigcup_{X \in \mathfrak{B}} X)^\perp = \bigcap_{X \in \mathfrak{B}} X^\perp$
774 and if $\mathfrak{B} \neq \emptyset$ then $(\bigcap_{X \in \mathfrak{B}} X)^\perp \supseteq (\bigcup_{X \in \mathfrak{B}} X^\perp)$ 4. $(\bigcup_{X \in \mathfrak{B}} X^{\perp\perp})^\perp = (\bigcup_{X \in \mathfrak{B}} X)^\perp$.

775 **Proof.** 1, 2, 3 are evident by definition. Observe that $(\bar{\Pi}, \bar{\Pi}, \perp)$ where $\perp = \{(p, q) \mid p \perp q\}$
776 is called a *polarity* on [5] and it defines a Galois connection [2]. For 4. observe that

$$777 \left(\bigcup_{X \in \mathfrak{B}} X^{\perp\perp}\right)^\perp = \bigcap_{X \in \mathfrak{B}} X^{\perp\perp\perp} = \bigcap_{X \in \mathfrak{B}} X^\perp = \left(\bigcup_{X \in \mathfrak{B}} X\right)^\perp.$$

778

779 ► **Proposition 34.** Let us consider $A, B, C \in \mathbb{P}$. Then:

780 1. $C \subseteq A \multimap B \iff C * A \subseteq B^{\perp\perp}$.

781 2. $C * A = \bigcap \{B \in \mathbb{B} \mid C \subseteq A \multimap B\} = \min\{B \in \mathbb{B} \mid C \subseteq A \multimap B\}$.

782 (In particular, if $B \in \mathbb{B}$ then $C \subseteq A \multimap B \iff C * A \subseteq B$).

783 **Proof.** 1. Observe that $A \multimap B = (A \bullet B^\perp)^\perp$.

$$\begin{aligned}
784 \quad C \subseteq (A \bullet B^\perp)^\perp &\Leftrightarrow \forall p \in C \quad \forall q \in A \quad \forall r \in B^\perp \quad p \perp q \bullet r \\
&\Leftrightarrow \forall p \in C \quad \forall q \in A \quad \forall r \in B^\perp \quad \bar{\nu}(p|q^1|r^2) \in \perp \\
&\Leftrightarrow \forall p \in C \quad \forall q \in A \quad \forall r \in B^\perp \quad (\nu\mathbb{N}^2)\left((\nu\mathbb{N}^1)(p|q^1|r^2)\right) \in \perp \\
&\Leftrightarrow \forall p \in C \quad \forall q \in A \quad \forall r \in B^\perp \quad \bar{\nu}\left((\nu\mathbb{N}^1)(p|q^1)^{-2}|r\right) \in \perp \\
&\Leftrightarrow \forall p \in C \quad \forall q \in A \quad \forall r \in B^\perp \quad p * q \perp r \\
&\Leftrightarrow C * A \subseteq B^{\perp\perp}
\end{aligned}$$

785 2. Applying the immediate previous result, we have that $C * A \subseteq C * A \iff C \subseteq A \multimap$
786 $(C * A)$. Then $C * A \in \{B \mid C \subseteq A \multimap B\}$. On the other hand, if $C \subseteq A \multimap B$, then
787 $C * A \subseteq B$.

788 \blacktriangleleft

789 **► Proposition 33.** For any $A, B \in \mathbb{P}$, it holds that $(A \bullet B)^\perp = (A^{\perp\perp} \bullet B^{\perp\perp})^\perp$. As a
790 consequence, $A \otimes B = A^{\perp\perp} \otimes B^{\perp\perp}$, $A \wp B = A^{\perp\perp} \wp B^{\perp\perp}$ and $A \multimap B = A^{\perp\perp} \multimap B^{\perp\perp}$.

791 **Proof.** Observe that $\bar{\nu}(p^1 | q^2 | r) \equiv \bar{\nu}(((\nu\mathbb{N}^2)(r | q^2))^{-1} | p) \equiv \bar{\nu}(((\nu\mathbb{N}^1)(r | p^1))^{-2} | q)$ for all
792 processes p, q, r .

793 The inclusion $(A^{\perp\perp} \bullet B^{\perp\perp})^\perp \subseteq (A \bullet B)^\perp$ is true by anti monotonicity of the orthogonal
794 operator. It remains to prove the reverse inclusion. Let us consider $r \in (A \bullet B)^\perp$ and try to
795 prove that $r \in (A^{\perp\perp} \bullet B^{\perp\perp})^\perp$. We know that $\bar{\nu}(((\nu\mathbb{N}^2)(r | q^2))^{-1} | p) \equiv \bar{\nu}(p^1 | q^2 | r) \in \perp$ for
796 all $p \in A, q \in B$ by the observation above. Then $((\nu\mathbb{N}^2)(r | q^2))^{-1} \in A^\perp = A^{\perp\perp\perp}$ and thus
797 $\bar{\nu}(p^1 | q^2 | r) \in \perp$ for all $p \in A^{\perp\perp}, q \in B$. Applying again the observation above we conclude
798 that $((\nu\mathbb{N}^1)(r | p^1))^{-2} \in B^{\perp\perp\perp}$ for all $p \in A^{\perp\perp}$. Finally again by this observation we get
799 $\bar{\nu}(p^1 | q^2 | r) \in \perp$ for all $p \in A^{\perp\perp}, q \in B^{\perp\perp}$, what we wanted to prove. \blacktriangleleft

800 **D** The induced conjunctive involutive monoidal algebra

801 **► Proposition 37.** The tuple $(\mathbb{B}, \subseteq, \otimes, (\cdot)^\perp)$, where (\mathbb{B}, \subseteq) is the lattice defined in Proposi-
802 tion 29, is a conjunctive involutive structure.

803 **Proof.** We have to check that the first axioms of conjunctive structures are satisfied. The
804 first part is just Proposition 29. Parts 2, 4 and 5 directly follow from the definition of \mathbb{B}, \otimes
805 and from Proposition 28. We prove the first case of part 3. For any $\mathfrak{B} \subseteq \mathbb{B}$, we have

$$\begin{aligned}
806 \quad A \otimes \bigvee_{B \in \mathfrak{B}} B &\stackrel{\text{def of } \vee}{=} A \otimes \left(\bigcup_{B \in \mathfrak{B}} B\right)^{\perp\perp} \stackrel{\text{Prop 33}}{=} A \otimes \bigcup_{B \in \mathfrak{B}} B \stackrel{\text{def of } \otimes}{=} (A \bullet \bigcup_{B \in \mathfrak{B}} B)^{\perp\perp} = \\
807 \quad &\left(\bigcup_{B \in \mathfrak{B}} (A \bullet B)\right)^{\perp\perp} \stackrel{\text{Prop 28.4}}{=} \left(\bigcup_{B \in \mathfrak{B}} (A \bullet B)^{\perp\perp}\right)^{\perp\perp} \stackrel{\text{def of } \otimes}{=} \left(\bigcup_{B \in \mathfrak{B}} (A \otimes B)\right)^{\perp\perp} \stackrel{\text{def of } \vee}{=} \bigvee_{B \in \mathfrak{B}} (A \otimes B)
\end{aligned}$$

810

811 \blacktriangleleft

812 Following we define a localized version for the application $*$ of Definition 31:

813 **► Definition 65.** Let us consider bijections $\iota_X : \mathbb{N} \rightarrow X$ and $\iota_Y : \mathbb{N} \rightarrow Y$ for (infinite) sets
814 of names X, Y . Let us define:

- 815 1. Given $i = 1, 2$ we denote by $i.X$ the set $\{\iota_X(n.i) \mid n \in \mathbb{N}\}$. The associated bijection between
816 \mathbb{N} and $i.X$ is $\iota_X \circ \iota_i : \mathbb{N} \rightarrow X$ and then $i.X \triangleq \text{Im}(\iota_X \circ \iota_i)$. Whenever it does not cause
817 confusion, we will write p^X instead of p^{ι_X} .
- 818 2. The fusion induced by ι_X, ι_Y is $\varepsilon_{X,Y} \triangleq \prod_{n \in \mathbb{N}} (\iota_X(n) \leftrightarrow \iota_Y(n))$.

819 3. Given processes $p, q \in \bar{\Pi}$ define $p \overset{\circledast}{X, Y} q \triangleq \left((\nu 1.X)(\nu Y)(p^X | q^Y | \underset{1.X, Y}{\overset{\varepsilon}{\cdot}}) \right)^{(-X).(-2)}$ and

820 $p \overset{\odot}{X, Y} q \triangleq \bar{\nu}(p^X | q^Y | \underset{X, Y}{\overset{\varepsilon}{\cdot}}).$

821 ► **Remark 66.** The following properties are immediate from Proposition 63:

822 1. We note that all of the above definitions depend on the bijections ι_X, ι_Y rather than
823 only on X, Y . The choice to use X and Y is to simplify the writing. Observe also that
824 $(\nu 1.X)(\nu Y)(p^X | q^Y | \underset{1.X, Y}{\overset{\varepsilon}{\cdot}}) = (p \overset{\circledast}{X, Y} q)^{2.X}$ and $\text{FN} \left((\nu 1.X)(\nu Y)(p^X | q^Y | \underset{1.X, Y}{\overset{\varepsilon}{\cdot}}) \right) \subseteq 2.X$.

825 2. If X, Y are disjoint, $p \in A \multimap B$ and $r \in A$, then $p \overset{\circledast}{X, Y} r \in B$.

826 3. If X, Y are disjoint, $p \in C$ and $q \in C^\perp$ then $p \overset{\odot}{X, Y} q \in \perp$.

827 ► **Lemma 67.** Let us consider $A, B, C \in \mathcal{P}(\bar{\Pi})$ and the following sets of names and injections:

$$\begin{aligned} X &\triangleq \mathbb{N}^1, & \iota_X &: \mathbb{N} \rightarrow X, & \iota_X &\triangleq \iota_1 \\ Y &\triangleq \mathbb{N}^{1.2.2}, & \iota_Y &: \mathbb{N} \rightarrow Y, & \iota_Y(n) &\triangleq n.1.2.2 \\ Z &\triangleq \mathbb{N}^{1.2}, & \iota_Z &: \mathbb{N} \rightarrow Z, & \iota_Z(n) &\triangleq n.1.2 \\ W &\triangleq \mathbb{N}^{2.2.2}, & \iota_W &: \mathbb{N} \rightarrow W, & \iota_W(n) &\triangleq n.2.2.2 \end{aligned}$$

829 Then $S_4 \triangleq \underset{1.X, Y}{\overset{\varepsilon}{\cdot}} | \underset{1.Z, 2.X}{\overset{\varepsilon}{\cdot}} | \underset{2.Z, W}{\overset{\varepsilon}{\cdot}} \in (A \multimap B) \multimap (B \multimap C) \multimap A \multimap C$.

830 **Proof.** Let us consider $p \in A \multimap B, q \in B \multimap C, r \in A, s \in C^\perp$. We must prove that
831 $\bar{\nu}(p^X | q^Z | r^Y | s^W | S_4) \in \perp$.

832 By Remark 66 we have $p \overset{\circledast}{X, Y} r \in B; q \overset{\circledast}{Z, 2.X} (p \overset{\circledast}{X, Y} r) \in C$ and $(q \overset{\circledast}{Z, 2.X} (p \overset{\circledast}{X, Y} r)) \overset{\odot}{2.Z, W} s \in \perp$.

833 This last process being structurally equivalent to $\bar{\nu}(p^X | q^Z | r^Y | s^W | S_4) \in \perp$, which ends
834 the proof.

835 ◀

836 ► **Lemma 68.** Let us consider $A, B, C \subseteq \mathcal{P}(\bar{\Pi})$ and the substitution $\tau : \mathbb{N}^1 \rightarrow \mathbb{N}^2$ defined
837 as $\tau(n.1.1.1) \triangleq n.1.2, \tau(n.2.1.1) \triangleq n.1.2.2$ and $\tau(n.2.1) \triangleq n.2.2.2$. Then $S_5 \triangleq \mathcal{E}_\tau \in$
838 $((A \otimes B) \otimes C) \multimap (A \otimes (B \otimes C))$.

839 **Proof.** By definition of \multimap and \otimes we have $((A \otimes B) \otimes C) \multimap (A \otimes (B \otimes C)) = \left(((A \bullet B) \bullet C) \bullet$

840 $(A \bullet (B \bullet C))^\perp \right)^\perp$. Let us consider $p \in A, q \in B, r \in C$ and $s \in (A \bullet (B \bullet C))^\perp$. We must prove

841 that $\bar{\nu}(\mathcal{E}_\tau | p^{1.1.1} | q^{2.1.1} | r^{2.1} | s^2) \in \perp$. By Proposition 63 we get $\bar{\nu}(\mathcal{E}_\tau | p^{1.1.1} | q^{2.1.1} | r^{2.1} | s^2) \equiv$

842 $(\nu \mathbb{N}^2)(p^{1.2} | q^{1.2.2} | r^{2.2.2} | s^2) \equiv \bar{\nu}(p^1 | q^{1.2} | r^{2.2} | s) \in \perp$; the last being consequence of

843 $p^1 | q^{1.2} | r^{2.2} \in A \bullet (B \bullet C)$. ◀

844 ► **Lemma 69.** Let us consider $A \subseteq \mathcal{P}(\bar{\Pi})$ and the substitution $\tau : \mathbb{N}^1 \rightarrow \mathbb{N}^2$ s.t. $\tau(n.1) \triangleq n.2.2$.
845 Then $S_6 \triangleq \mathcal{E}_\tau \in A \multimap (1 \otimes A)$.

846 **Proof.** By definition of \multimap and \otimes we have $A \multimap (1 \otimes A) = \left(A \bullet (1 \bullet A)^\perp \right)^\perp$. Let us consider

847 $p \in A$ and $q \in (1 \bullet A)^\perp$. It suffices to prove that $\varepsilon_\tau \perp p^1 | q^2$, i.e. $(\bar{\nu})(\varepsilon_\tau | p^1 | q^2) \in \perp$. Applying

848 Proposition 63 we get $(\bar{\nu})(\varepsilon_\tau | p^1 | q^2) \equiv_\alpha (\nu \mathbb{N}^2)(q^2 | p^{2.2}) \equiv_\alpha (\nu \mathbb{N}^2)(q^2 | \perp | p^{2.2}) \equiv_\alpha (\bar{\nu})(q | \perp | p^2)$.

849 Since $\perp | p^2 \in 1 \otimes A$ and $q \in (1 \otimes A)^\perp$, then $(\bar{\nu})(q | \perp | p^2) \in \perp$. ◀

850 ► **Lemma 70.** Let us consider $A \subseteq \mathcal{P}(\bar{\Pi})$ and the substitution $\tau : \mathbb{N}^1 \rightarrow \mathbb{N}^2$ s.t. $\tau(n.1) \triangleq n.2.2$.

851 Then $S_6 \triangleq \mathcal{E}_\tau \in A \multimap (1 \otimes A)$.

852 **Proof.** By definition of \multimap and \otimes we have $(1 \otimes A) \multimap A = \left((1 \bullet A) \bullet A^\perp \right)^\perp$. Let us consider
 853 $p \in A$ and $q \in A^\perp$. It suffices to prove that $\varepsilon_\tau \perp \mathbb{1} \mid p^{2.1} \mid q^2$, i.e. $(\bar{\nu})(\varepsilon_\tau \mid \mathbb{1} \mid p^{2.1} \mid q^2) \in \perp$.
 854 Applying Proposition 63 we get $(\bar{\nu})(\varepsilon_\tau \mid \mathbb{1} \mid p^{2.1} \mid q^2) \equiv_\alpha (\nu \mathbb{N}^2)(p^2 \mid q^2) \equiv_\alpha (\bar{\nu})(p \mid q)$, being the
 855 last at \perp because $p \perp q$ by assumption. \blacktriangleleft

856 E Conjunctive algebras for concurrency

857 **► Proposition 42.** Let us consider $(\mathbb{C}, \otimes, (\cdot)^\perp, \mathbb{1}, F, \preceq)$ a CS with composition F . Then
 858 $aFb \preceq c \iff a \preceq b \triangleright_F c$ for all $a, b, c \in \mathbb{C}$.

859 **Proof.** \implies $aFb \preceq c$ implies that $a \in \{x \in \mathbb{C} \mid xFb \preceq c\}$ and then $a \preceq b \triangleright_F c$.

860 \impliedby Since F is monotone, we have

$$861 \quad aFb \preceq (b \triangleright_F c)Fb = \left(\bigvee \{x \in \mathbb{C} \mid xFa \preceq c\} \right) \mid b \preceq \bigvee \{xFb \mid x \in \mathbb{C}, xFb \preceq c\} \preceq c \quad \blacktriangleleft$$

862 **► Lemma 71.** Let us consider $\mathfrak{B} \subseteq \mathbb{P} = \mathcal{P}(\bar{\Pi})$ and $A \subseteq \mathbb{P}$. Then $(\bigvee \mathfrak{B}) \mid A \subseteq \bigvee \{B \mid A \mid B \in \mathfrak{B}\}$
 863 $\mathfrak{B}\} = ((\bigcup \mathfrak{B}) \mid A)^\perp$.

864 **Proof.** Let us consider $(P, e) \in \bigvee \mathfrak{B}$, $(Q, f) \in A$ and $(H, h) \in ((\bigcup \mathfrak{B}) \mid A)^\perp$. It suffices to
 865 prove that $\bar{\nu}(P \mid Q \mid H, efh) \in \perp$. By the hypothesis that $(P, e) \in \bigvee \mathfrak{B} = (\bigcup \mathfrak{B})^\perp$ it suffices
 866 to prove that $(Q \mid H, fh) \in (\bigcup \mathfrak{B})^\perp$. Let us consider that $(R, r) \in \bigcup \mathfrak{B}$. Then, it suffices
 867 to prove that $\bar{\nu}(H \mid R \mid Q, hrq) \in \perp$. This sentence is true because $(R \mid Q, rf) \in (\bigcup \mathfrak{B}) \mid A$ and
 868 $(H, h) \in ((\bigcup \mathfrak{B}) \mid A)^\perp$. \blacktriangleleft

869 **► Proposition 43.** Consider the CS $(\mathbb{B}, \otimes, (\cdot)^\perp, \mathbb{1}, \subseteq)$ of behaviors over $\bar{\Pi}$.

870 1. The operation \parallel (from Definition 32) is a composition over \mathbb{C} , the equivalence \equiv_{\parallel} is
 871 equality and its right adjoint is $B \triangleright_{\parallel} C := (B \parallel C^\perp)^\perp$.

872 2. The operation \otimes (from Definition 32) is a composition over \mathbb{C} and its right adjoint is
 873 $B \triangleright_{\otimes} C := (((B^2 \parallel C^\perp)^\perp)_{\uparrow \mathbb{N}^1})^{-1}$.

874 **Proof.**

875 1. Since $(\bar{\Pi}, \mid, \mathbb{1})$ is an abelian monoid, then $(\mathbb{B}, \parallel, \mathbb{1})$ is also an abelian monoid with unite
 876 $\mathbf{1} := \{\mathbb{1}\}^\perp$. By definition, \parallel is an increasing function. Observe that $(\bigvee \mathfrak{B}) \mid A \subseteq$
 877 $\bigvee \{B \parallel A \mid B \in \mathfrak{B}\}$ for all $\mathfrak{B} \subseteq \mathbb{B}$ and $A \in \mathbb{B}$ by Lemma 71. The continuity in the right
 878 argument is proven symmetrically. On the other hand, $A \parallel B \subseteq C \iff C^\perp \subseteq (A \parallel B)^\perp$, which
 879 is equivalent to $\forall p \in A, q \in B, r \in C^\perp \quad \bar{\nu}(p \mid q \mid r) \in \perp$. The last condition is equivalent to
 880 $A \subseteq (B \parallel C^\perp)^\perp$, which proves that $B \triangleright_{\parallel} C := (B \parallel C^\perp)^\perp$ is the right adjoint of \parallel .

881 2. By Definition 1 we have $(\bigvee \mathfrak{B}) \otimes A = \bigvee \{B \otimes A \mid B \in \mathfrak{B}\}$ and $A \otimes (\bigvee \mathfrak{B}) = \bigvee \{A \otimes B \mid B \in \mathfrak{B}\}$.

882 By definition, \otimes is an increasing function. On the other hand, $A \otimes B \subseteq C \iff C^\perp \subseteq$
 883 $(A \otimes B)^\perp$, which is equivalent to $\forall p \in A, q \in B, r \in C^\perp \quad \bar{\nu}(p^1 \mid q^2 \mid r) \in \perp$. The last condition
 884 is equivalent to $A^1 \subseteq ((B^2 \mid C^\perp)^\perp)_{\uparrow \mathbb{N}^1} \iff A \subseteq (((B^2 \mid C^\perp)^\perp)_{\uparrow \mathbb{N}^1})^{-1}$, which proves that

885 $B \triangleright_{\otimes} C := (((B^2 \mid C^\perp)^\perp)_{\uparrow \mathbb{N}^1})^{-1}$ is the right adjoint of \otimes .

886 \blacktriangleleft

F

 Embedding the π -calculus through Honda-Yoshida's combinators

► **Definition 72.** Let us consider a pole $\perp \subseteq \bar{\Pi}$. We say that \perp is regular iff for all $(P, e), (Q, f)$ (PGF)-processes and names u, \bar{x} , $\{(P \mid Q, ef)\}^\perp \subseteq \{(u^\dagger \langle \bar{x} \rangle . P \mid v^\dagger \langle \bar{x} \rangle . Q, ef)\}^\perp$ whenever $u \underset{ef}{\sim} v$.

The combinators of Honda & Yoshida are also encoded into PGF by usual π -calculus (pure) processes:

► **Definition 73.** Let us define the following processes:

$$\begin{array}{l} \mathbf{m}(a, x) \triangleq a^\dagger \langle x \rangle \\ \mathbf{k}(a) \triangleq a^\dagger(x) \\ \mathbf{f}(a, b) \triangleq a^\dagger(x).b^\dagger \langle x \rangle \\ \mathbf{bl}(a, b) \triangleq a^\dagger(x).f(x, b) \end{array} \quad \left| \quad \begin{array}{l} \mathbf{br}(a, b) \triangleq a^\dagger(x).f(b, x) \\ \mathbf{d}(a, b, c) \triangleq a^\dagger(x).(m(b, x) \mid m(c, x)) \\ \mathbf{s}(a, b, c) \triangleq a^\dagger(x).f(b, c) \end{array} \right.$$

► **Remark 74.** Let us consider a regular pole \perp . Then for all names a, b, c, d, x we have:

$$\begin{array}{l} \{\mathbb{1}\}^\perp \subseteq \{\mathbf{k}(a) \mid m(a, x)\}^\perp \\ \{m(b, x)\}^\perp \subseteq \{\mathbf{f}(a, b) \mid m(a, x)\}^\perp \\ \{m(b, x) \mid m(c, x)\}^\perp \subseteq \{\mathbf{d}(a, b, c) \mid m(a, x)\}^\perp \end{array} \quad \left| \quad \begin{array}{l} \{f(x, b)\}^\perp \subseteq \{\mathbf{bl}(a, b) \mid m(a, x)\}^\perp \\ \{f(b, x)\}^\perp \subseteq \{\mathbf{br}(a, b) \mid m(a, x)\}^\perp \\ \{f(b, c)\}^\perp \subseteq \{\mathbf{s}(a, b, c) \mid m(a, x)\}^\perp \end{array} \right.$$

► **Proposition 48.** Let us consider a CIMA $(\mathbb{C}, \preceq, \otimes, \parallel, (\cdot)^\perp, 1, \mathcal{S})$ whose pole \perp is regular. Then \mathbb{C} is a HYA with the definition $\mathbf{M}(a, b) \triangleq \{(a^\dagger \langle b \rangle . \mathbb{1}, \Delta_{\mathbb{N}})\}^{\perp\perp}$.

Proof. By definition, $\mathbf{M}(a, b) \in \mathcal{S}$ because is not empty. The other combinators $\mathbf{K}(a), \mathbf{F}(a, b), \mathbf{Bl}(a, b), \mathbf{Br}(a, b), \mathbf{D}(a, b, c)$ and $\mathbf{S}(a, b, c)$ are given by Definition 46 and they are inhabited by their respective encoding on (PGF). For instance: By the Remark 74 we have $\{\mathbb{1}\}^\perp \subseteq \{\mathbf{k}(a) \mid m(a, x)\}^\perp = \left(\{\mathbf{k}(a)\}^{\perp\perp} \mid \mathbf{M}(a, x)\right)^\perp = \left(\{\mathbf{k}(a)\}^{\perp\perp} \parallel \mathbf{M}(a, x)\right)^\perp$. Applying the orthogonal map we get: $\{\mathbf{k}(a)\}^{\perp\perp} \parallel \mathbf{M}(a, x) \subseteq \{\mathbb{1}\}^{\perp\perp} =: 1$.

By Proposition 42 we deduce that $\{\mathbf{k}(a)\}^{\perp\perp} \subseteq \mathbf{M}(a, x) \triangleright 1$ for all a, b, x . Taking the meet on x we get $\mathbf{k}(a) \in \{\mathbf{k}(a)\}^{\perp\perp} \subseteq \bigwedge_{x \in \mathbb{N}} (\mathbf{M}(a, x) \triangleright 1) = \mathbf{K}(a)$ and thus $\mathbf{k}(a) \in \mathbf{K}(a)$.

For $\mathbf{d}(a, b, c)$ the reasoning is almost the same: By Remark 74 we have $\{m(b, x) \mid m(c, x)\}^\perp \subseteq \{\mathbf{d}(a, b, c) \mid m(a, x)\}^\perp$. This inclusion can be rewritten as $\left(\mathbf{M}(b, x) \parallel \mathbf{M}(c, x)\right)^\perp \subseteq \left(\{\mathbf{d}(a, b, c)\}^{\perp\perp} \parallel \mathbf{M}(a, x)\right)^\perp$. Applying the orthogonal map we get $\{\mathbf{d}(a, b, c)\}^{\perp\perp} \parallel \mathbf{M}(a, x) \subseteq \mathbf{M}(b, x) \parallel \mathbf{M}(c, x)$ for all names a, b, c, x , which is equivalent to $\{\mathbf{d}(a, b, c)\}^{\perp\perp} \subseteq \mathbf{M}(a, x) \triangleright \left(\mathbf{M}(b, x) \parallel \mathbf{M}(c, x)\right)$. Again taking the meet on x we conclude that $\mathbf{d}(a, b, c) \in \mathbf{D}(a, b, c)$.

For the rest of the combinators the proof is entirely analogous. ◀

G

 Parallelism cannot be derived

We give here detailed proof for the results given in of Section 6. First, as was noticed by Munch-Maccagnoni [16], we can characterize realizers as follows:

► **Proposition 75.** For any formula A , $t \Vdash A$ iff $\forall u. (u \Vdash A^\perp \Rightarrow \langle t \parallel u \rangle \in \perp)$.

Proof. This is a direct consequence of the observation that $|A^\perp| = |A|^\perp$. ◀

The following lemma, which is barely more than a simple observation, describe on realizers for the implication.

919 ► **Lemma 76.** For any closed formulas with parameters A, B , we have: $t \Vdash A \multimap B$ iff for
 920 any $u \Vdash A$, $v \Vdash B^\perp$, $\langle t \parallel (u, v) \rangle \in \perp$.

921 **Proof.** By definition, we have:

$$\begin{aligned}
 922 \quad |A \multimap B| &= |(A \otimes B^\perp)^\perp| \\
 923 \quad &= |(A \otimes B^\perp)|^\perp \\
 924 \quad &= \{(u, v) : u \in |A| \wedge v \in |B^\perp|\}^{\perp\perp\perp} \\
 925 \quad &= \{(u, v) : u \in |A| \wedge v \in |B^\perp|\}^\perp \\
 926
 \end{aligned}$$

927 Hence the result. ◀

928 Finally, to simplify the proofs below, let us recall that terms of the λ -calculus can be
 929 encoded in L using the following shorthands:

$$\begin{array}{c}
 930 \quad A \multimap B \triangleq A^\perp \wp B \quad \left| \quad \lambda\kappa.t \triangleq \mu(\kappa, \kappa').\langle t \parallel (\kappa, \kappa') \rangle \quad \kappa' \notin \text{FV}(t) \\
 \quad u \cdot e \triangleq (u, e) \quad \left| \quad t u \triangleq \mu\kappa.\langle t \parallel u \cdot \kappa \rangle \quad \kappa \notin \text{FV}(t, u)
 \end{array}$$

These shorthands are sound with respect to computation (in the sense that the β -reduction of the λ -calculus is simulated, with an evaluation strategy dictated by the polarities of A and B (call-by-name corresponds to both formulas being negative, while call-by-value corresponds to both being positive)) but also to types. Indeed, one can show that the following rules are admissible³:

$$\frac{\vdash t : B \mid \kappa : A^\perp, \Gamma}{\vdash \lambda\kappa.t : A \multimap B \mid \Gamma} \text{ (abs)} \qquad \frac{\vdash t : A \multimap B \mid \Gamma \quad \vdash u : A \mid \Gamma}{\vdash t u : B \mid \Gamma} \text{ (app)}$$

931 ► **Proposition 51.** The tuple $(\mathbf{H}, \subseteq, \otimes, (\cdot)^\perp, \perp, \mathcal{S})$, where $\perp \triangleq \mathcal{T}_+^0$ and $\mathcal{S} = \mathbf{H} \setminus \emptyset$, is a CIMA.

932 **Proof.** We first show that $(\mathbf{H}, \subseteq, \otimes, (\cdot)^\perp)$ is a conjunctive structure.

- 933 ■ The set of behaviors, ordered by inclusion, defines a complete lattice whose join is given
- 934 by the union.
- 935 ■ The map $\cdot \otimes \cdot$ is clearly covariant in its two arguments.
- 936 ■ Similarly, it is clear that $(\cdot)^\perp$ is anti-monotonic.
- 937 ■ As is usual with orthogonality relations, we have that for any $\mathcal{U} \subset \mathcal{T}_+^0$ (resp. \mathcal{T}_-^0):

$$938 \quad \left(\bigcup_{T \in \mathcal{U}} T \right)^\perp = \bigcap_{T \in \mathcal{U}} T^\perp$$

939 Observing that if for any set $\mathcal{U} \subset \mathcal{T}^0$ such that $\mathcal{U} \cap \mathcal{T}_+^0 \neq \emptyset$ and $\mathcal{U} \cap \mathcal{T}_-^0 \neq \emptyset$ we have

$$\begin{aligned}
 940 \quad \left(\bigcup_{T \in \mathcal{U}} T \right)^\perp &= \left(\bigcup_{T_+ \in \mathcal{U} \cap \mathcal{T}_+} T_+ \cup \bigcup_{T_- \in \mathcal{U} \cap \mathcal{T}_-} T_- \right)^\perp \\
 941 \quad &= \left(\bigcup_{T_+ \in \mathcal{U} \cap \mathcal{T}_+} T_+ \right)^\perp \cap \left(\bigcup_{T_- \in \mathcal{U} \cap \mathcal{T}_-} T_- \right)^\perp \\
 942 \quad &= \bigcap_{T_+ \in \mathcal{U} \cap \mathcal{T}_+} T_+^\perp \cap \bigcap_{T_- \in \mathcal{U} \cap \mathcal{T}_-} T_-^\perp = \bigcap_{T \in \mathcal{U}} T^\perp \\
 943
 \end{aligned}$$

944 (In fact, all these sets are empty since $\mathcal{T}_+ \cap \mathcal{T}_- = \emptyset$)

³ They are in fact derivable up to weakening.

945 ■ The map $(\cdot)^\perp$ is by definition (of \mathbf{H}) involutive on \mathbf{H} .

946 We can now prove that, when considering the unit $\mathbb{1} \triangleq \mathcal{T}_+^0$ (which is indeed a behavior), the
 947 set of non-empty behaviors $\mathcal{S} = \mathbf{H} \setminus \emptyset$ defines a valid separator. This mostly amounts to
 948 finding a realizer for each of the different axioms.

949 ■ \mathcal{S} is clearly upwards-closed.

950 ■ \mathcal{S} is compatible with modus ponens. Indeed, if $A, B \in \mathbf{H}$ are such that $A \multimap B \in \mathcal{S}$ and
 951 $A \in \mathcal{S}$, by definition of \mathcal{S} it means that there exists two terms t and u such that $t \in A \multimap B$
 952 and $u \in A$. Then $t u = \mu\kappa.\langle t \parallel (u, \alpha) \rangle \Vdash B$, using the typing rule for application above
 953 and Proposition 50.

954 ■ $\mu(x, \alpha).\langle x \parallel \mu(\kappa_1, \kappa_2).\langle (\kappa_2, \kappa_1) \parallel \alpha \rangle \rangle \Vdash (A \otimes B) \multimap (B \otimes A)$

955 ■ $\lambda\alpha.\mu(x, \beta).\langle x \parallel \mu(\kappa, \kappa').\langle (\alpha \kappa, \kappa') \parallel \beta \rangle \rangle \Vdash (A \multimap B) \multimap (A \otimes C) \multimap (B \otimes C)$

956 ■ $\mu(x, \alpha).\langle x \parallel \mu(y, \kappa_3).\langle y \parallel \mu(\kappa_1, \kappa_2).\langle (\kappa_1, (\kappa_2, \kappa_3)) \parallel \alpha \rangle \rangle \rangle \Vdash ((A \otimes B) \otimes C) \multimap (A \otimes (B \otimes C))$

957 ■ For any $t \in \mathcal{T}_+^0$, $\lambda\kappa.(t_+, \kappa) \Vdash A \multimap \mathbb{1} \otimes A$

958 ■ $\mu(x, \kappa).\langle x \parallel \mu(_, \kappa').\langle \kappa' \parallel \kappa \rangle \rangle \Vdash \mathbb{1} \otimes A \multimap A$

959 ■ $\mu(x, \alpha).\langle x \parallel \mu(\kappa, \kappa').\langle (\kappa, \kappa') \parallel \alpha \rangle \rangle \Vdash (A \multimap B) \multimap (B^\perp \multimap A^\perp)$

960 For each of the last terms, it suffices to check that they have the corresponding type and
 961 then use adequacy (cf. Proposition 50). ◀