A Spatial Logic for Concurrency (Part II)

Luís Caires
Departamento de Informática
FCT/UNL, Portugal

Luca Cardelli
Microsoft Research
Cambridge, UK

Abstract

We present a modal logic for describing the spatial organization and the behavior of distributed systems. In addition to standard logical and temporal operators, our logic includes spatial operations corresponding to process composition and name hiding, and a fresh name quantifier. Properties of concurrent systems can also be defined by second-order quantification and hence (through an encoding) by recursion. A central aim of our logic is the combination of a notion of freshness with inductive and coinductive definitions of properties.

1 Introduction

1.1 Spatial Logics

We develop a logic to describe properties of distributed concurrent systems, for specification and model-checking purposes; we believe that the peculiar characteristics of such systems justify the introduction of new logical constructs. Our first emphasis is on *distributed* systems, meaning that we should be able to talk about properties of distinct subsystems, such as subsystems that reside at different locations, and subsystems that privately share hidden resources. For this purpose, we introduce *spatial* (as opposed to temporal) logical operators; for example, we may talk about a property holding somewhere (as opposed to sometimes). Our second emphasis is on *concurrent* systems: we want a logic that unambiguously talks about concurrency and (nowadays) privacy. For this purpose, the intended model of our logic is built explicitly from a standard process calculus. Our formulas denote collections of processes subject to certain closure conditions, with some logical operators mapping directly to process composition and name hiding. In Part I of this paper [2] we study this intended model, which is used here to establish the soundness of the logical rules. The central focus of this Part II, however, is proof theory, including a new presentation style for the rules of spatial logics, and a proof of cut-elimination for the first-order fragment.

Our logic is presented here as a modal sequent calculus with ”world constraints”. A formula in our logic describes a property of a particular part of a concurrent system at a particular time; therefore it is modal in space as well as in time. Structurally, the logic consists of a collection of left-right rules for logical operators, including essentially the standard rules of classical sequent calculus, plus the ones for temporal and spatial operators. In addition, there are special rules about the worlds: they add meaning to the logical operators, without interfering very much with the basic left-right structure of the sequent calculus. This added meaning can capture deep knowledge about the worlds; in our case, deep properties of process calculi.

We present the logic monolithically, since the intended expressiveness requires a well-integrated combination of features. But, in retrospect, we could imagine building the logic up in stages, e.g. for simpler application domains. In this incremental sense, the starting point can be considered to be Alex Simpson’s presentation of modal logic as a sequent calculus [14]. In the sequents of that calculus, formulas are indexed by the worlds they predicate over, so a sequent can talk about many distinct words at once. Each sequent includes also a finite set of reachability constraints over the worlds. Beyond minimal modal logic, various modal logics can be flexibly captured by additional ”rules about the worlds”, in the sense mentioned above.

$$\langle S\rangle\, \Gamma \vdash \Delta, \text{ that is: } \langle S\rangle\, u_1 : A_1, \ldots, u_n : A_n \vdash v_1 : B_1, \ldots, v_m : B_m$$

A_i, B_i	Formulas
u_i, v_j	Indexes, members of a process algebra (the worlds)
S	finite set of constraints (e.g. equations, reductions) over the algebra of the worlds

Figure 1: Sequents

We can enrich that basic logical structure for our purposes. The first natural step, as in many logics of concurrency, is to interpret reachability as reduction in a process calculus. Still, up to this point, the worlds are usually uninterpreted. In order to capture the spatial properties we are after, we impose further structure on the worlds and on the constraints: one of our typical constraints says that a world is made of the composition of two other worlds. In general, the additional world and constraint structure can be fashioned as an algebra; in our full logic it is a relatively complex process algebra.

The fragment of our logic that deals with process composition is relatively straightforward: composition shows up in the logic as a tensor, which is strongly related to linear connectives. The sequent-style presentation of this fragment should look relatively familiar, except for the constraints part. The relevant constraints are essentially constraints over a (concurrency) monoid, with some specific interactions with reduction. Along these lines, we could also easily add an explicit structure of locations to the process calculus, and related logical operators, as done in [4].

Far less obvious is what to do about hiding of private resources, which is represented in π-calculus by the name hiding operator. The hiding of a name in a process should correspond, logically, to a "hiding quantifier" that binds a private name in a formula; such a formula could then describe the use of that private name in the process. The study of such a quantifier, from a logical point of view, was started in [3, 1], and later independently in [5]. Our current understanding is that it is best to decompose such a hiding quantifier into two operators. The first one is a modal version of the fresh quantifier of Gabbay and Pitts [7]. This quantifier binds a fresh name in a formula; it can be motivated in very general terms and, besides, some notion of freshness is clearly relevant to the idea of hiding. The second one is a logical operator (not a quantifier), called *revelation* [5], that relates to name hiding in strong analogy to the way tensor relates to process composition. A simple combination of fresh quantification and revelation then yields hiding, in the intuitive sense that if something is hidden, we can choose to name it (reveal it) by any name that is fresh.

Many natural examples of use of our logic involve recursive formulas. As in μ-calculus [9], recursion may be encapsulated into (temporal or spatial) box and diamond modalities, but it is best to have it in full power. Two typical examples of recursion that attract us in our context are: (1) a process having an arbitrary number of hidden resources, and (2) a process generating an infinite supply of fresh names. Particularly, the interaction of recursion and freshness is semantically quite challenging, and was investigated in Part I. In this paper we go one step further and introduce second-order quantification in the logic, from which we can define least and greatest fixpoints of formulas, almost along standard lines [12]. This generalization has some technical advantages.

1.2 Guide to Part I

We now give a brief overview of the contents of Part I [2]. A superficial understanding of these basic concepts should help intuitions; a deeper understanding is required to show soundness of the logic.

We start with a fixed process calculus (we use asynchronous π-calculus); let $\mathcal{P}$ be the collection of such processes. A *property* is a set of processes; a subset of $\mathcal{P}$. A closed formula denotes a property, namely, it denotes the collection of processes satisfying that formula. A property must be closed under structural equivalence of processes, and under a finite support condition [7], related to the fact that a formula can mention only a finite set of names.

The collection of all properties has the structure of a Boolean algebra under set inclusion, so we naturally get propositional connectives in the logic. The collection of all properties has also the structure of a commutative quantale, due to the parallel composition operator over processes; this induces the basic spatial connectives of the logic. Other process operators induce further spatial connectives.

1.3 Guide to Part II

In Section 2 we introduce the π-algebra that is used in the constraints and indexes of our sequents. A π-algebra is an abstraction of π-calculi, incorporating most of the characteristic properties of composition and hiding. Transpositions (*i.e.*, name swappings, as in nominal logic [11]) are an explicit part of the π-algebra; they are used to deal with open terms, and are also crucial in expressing some important rules ((S$\boldsymbol{\nu\nu}$), Figure 10). This algebra has two sorts: names and processes; the processes are actually called indexes, because they index formulas in sequents.

The algebra consists, as usual, of a collection of operators over a set of variables. However, one of the operators corresponds to π-calculus name hiding, so if we write $(\boldsymbol{\nu} x)\mathcal{X}$, both x and $\mathcal{X}$ are algebraic variables, and in particular x is a free variable of this algebraic term, and not a bound name as in π-calculus processes. An interpretation of this algebraic term into π-calculus would map x to a name n and $\mathcal{X}$ to a process P, to obtain a process $(\boldsymbol{\nu} n)P$ where n is a bound name.

Four relations, called constraints, are defined over the π-algebra, and are used in the constraint part of sequents. These relations are equality of indexes (incorporating π-calculus equations), reduction of indexes (incorporating π-calculus process reductions), and apartness of names and sets, much as in nominal logic.

In Section 3 we introduce our sequent calculus, which can be shown sound by an interpretation in the model of Part I [2]. We start with the syntax of formulas, which includes classical predicate logic, but also includes several non-standard connectives. In Part I we gave a satisfaction relation that interprets these new connectives as operations on processes. For example, the formula $A|B$ maps to process composition $P|Q$, in the sense that a process R satisfies $A|B$ if R has the form $P|Q$ with P satisfying A and Q satisfying B. In this paper, however, the meaning of $A|B$ is captured by its sequent calculus rules, which are essentially those of a tensor. We highlight here the left and right rules for $A|B$, which include many of the interesting features of our sequents:

$$\frac{\langle S\rangle\,\Gamma \vdash v : A, \Delta \quad \langle S\rangle\,\Gamma \vdash t : B, \Delta \quad u \doteq_S v|t}{\langle S\rangle\,\Gamma \vdash u : A|B, \Delta}\ (|\mathrm{R})$$

$$\frac{\begin{array}{c}[\mathcal{X} \textit{ and } \mathcal{Y} \textit{ not free in the conclusion}]\\ \langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\,\Gamma, \mathcal{X} : A, \mathcal{Y} : B \vdash \Delta\end{array}}{\langle S\rangle\,\Gamma, u : A|B \vdash \Delta}\ (|\mathrm{L})$$

The ($|$R) rule says: if we can show that index v satisfies formula A (i.e, that A holds at world v, written $v : A$), and that t satisfies B, and if we can show from the constraints in S that $u \doteq v|t$, then we can conclude that u satisfies $A|B$. Hence, the reading of this logical rules incorporates much of the intended satisfaction semantics [14]. The ($|$L) rule features the assumption ”$\mathcal{X}$ and $\mathcal{Y}$ not free in the conclusion (of the rule)” where $\mathcal{X}$ and $\mathcal{Y}$ are two algebraic variables of sort index. This assumption means, in particular, that $\mathcal{X}$ and $\mathcal{Y}$ are completely generic and unconstrained (*cf.* eigenvariables). A reading is: to show that $u : A|B$ entails Δ, we must show that for an arbitrary decomposition of u as $\mathcal{X}|\mathcal{Y}$, we have that $\mathcal{X} : A$ and $\mathcal{Y} : B$ entail Δ.

Composition also has a number of structural rules (”rules about the world”, as mentioned above in 1.1). Here is a simple one:

$$\frac{\langle S, u \doteq \mathbf{0}\rangle\, \Gamma \vdash \Delta \quad u|v \doteq_S \mathbf{0}}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\mathrm{S}|\mathbf{0})$$

Note that these world rules do not involve the logical connectives (we have $\Gamma \vdash \Delta$ above and below), and instead affect the $\langle S\rangle$ part. In most process calculi we have that if $u|v \equiv 0$ then both $u \equiv 0$ and $v \equiv 0$. This property does not derive from (|L) and (|R), but is embedded in (S|**0**). The rule reads as follows: if we can already infer from the $\langle S\rangle$ part of the constraints that $u|v \doteq_S \mathbf{0}$, and we have an additional constraint that $u \doteq_S \mathbf{0}$, that constraint is redundant and we can remove it. In this style, we can incorporate many peculiar properties of process calculi as world rules; many such rules analyze the consequences of an equation between two spatial operators (above, | vs. **0**), and are listed in Figure 10. All such rules have a similar reading in terms of eliminating "redundant" constraints.

In Section 4 we investigate proof theory, and in particular cut elimination for the first-order fragment of our logic. Because of the regular left-right structure of our rules, cut elimination falls largely along predictable lines; the indexes do not hinder, and rules such as (S|**0**) can be dealt with separately. The main difficulty is in the cut elimination case for the freshness quantifier. As in nominal logic, the result depends on an "equivariance" property of the logic [11], which is used to perform an α-conversion of fresh names over a whole derivation. Equivariance is embedded, in our case, in the (τ) rule in Figure 9, whose soundness depends on the main theorem of Part I. Expressing this rule in the general case of open formulas, requires introducing explicit transpositions over formulas, which entail some technical complications.

In Section 5 we go through a set of basic examples, to illustrate the expressive power of the logic.

In the Appendix, we collect proofs of results and technical details.

1.4 Results and Related Work

In this paper we present a logic for concurrency in a novel style based on a modal sequent calculus. We regularize and generalize the logics introduced in [1, 4, 5]. We show soundness of our inference rules in the π-calculus based model developed in Part I [2]. We prove a cut-elimination result for the first-order fragment, including cut-elimination for a fresh name quantifier (*cf.* Nominal Logic [11]).

A logic for processes including spatial operators was developed by Luís Caires in [1]. This logic introduced a hiding quantifier, but a satisfactory semantic treatment for it was not achieved until the contributions of [5, 2].

Andy Gordon was a coauthor with Luca Cardelli of initial versions of spatial logics [4, 5]. The present paper contains the first presentation of such a logic as a proper sequent calculus. Moreover, we now target the logic towards a more standard π-calculus, as opposed to the Ambient Calculus, but this is a minor point.

There are obvious connections between our basic spatial operators and linear logic; some have been explored in [4]. Our other logical operators, though, go beyond linear logic, particularly to deal with name hiding.

The first main difference between our logic and standard logics of concurrency (e.g. [8]) is the presence in our case of a tensor operator that corresponds to process composition. Usually, those other logics require formulas to denote processes up to bisimulation, which is difficult to reconcile with a tensor operator that can make distinctions between bisimilar processes (however, such an operator was anticipated by Dam [6]). In our case, we only require formulas to denote processes up to structural equivalence [2], so that a tensor operator makes easy sense. Sangiorgi has shown, for a closely related logic, that the equivalence induced by the logic is then essentially structural equivalence [13].

We had many illuminating conversations with Peter O'Hearn about Bunched Logic [10] and the formalization of logics for "rich" worlds. Beyond the dissimilarities due to our respective application domains, many themes are shared in our respective work. In both cases we have a tensor operator that must interact with an intuitionistic or classical conjunction. This interaction is handled by "bunched" contexts in the case of Bunched Logic, and by many-world sequents in our case.

The work of Gabbay and Pitts on the freshness quantifier happened at exactly the time we needed it [7], and has become central to this work.

We discussed above Alex Simpson's work, whose Ph.D. thesis we recently became aware of. Both his logic and the general organization of our logic seem related to labeled deduction systems [16], at least superficially; we are still investigating the connections.

2 Preliminaries

In this Section we introduce π-algebras, and constraint theories over the term π-algebra.

2.1 π-algebras

Definition 2.1 (π-algebra) *A π-algebra is a structure*

$$\Pi = \langle \mathcal{L}, \mathcal{P}, \mathbf{0}, |, \boldsymbol{\nu}, (\leftrightarrow)_{\mathcal{L}}, (\leftrightarrow)_{\mathcal{P}} \rangle$$

such that $\mathcal{L}$ is a countable set of labels (ℓ)*, $\mathcal{P}$ is a set of* processes (P, Q, R) *and*

- $\mathbf{0}$ *(void) is a distinguished process in $\mathcal{P}$*
- $-|-$ *(*composition*) is an operation $\mathcal{P} \times \mathcal{P} \to \mathcal{P}$*
- $(\boldsymbol{\nu}-)-$ *(*name hiding*, a.k.a. restriction) is an operation $\mathcal{L} \times \mathcal{P} \to \mathcal{P}$*
- $(-\leftrightarrow-)_{\mathcal{L}}-$ *(*transposition on labels*) is an operation $\mathcal{L} \times \mathcal{L} \times \mathcal{L} \to \mathcal{P}$*
- $(-\leftrightarrow-)_{\mathcal{P}}-$ *(*transposition on processes*) is an operation $\mathcal{L} \times \mathcal{L} \times \mathcal{P} \to \mathcal{L}$*

We will refer to the $\mathcal{L}$ part of a π-algebra Π by $\Pi_{\mathcal{L}}$, and likewise for the remaining components (*e.g.*, $\Pi_{\mathcal{P}}$). For instance, the asynchronous π-calculus $\mathrm{A}\pi$ is the π-algebra where $\mathrm{A}\pi_{\mathcal{L}}$ is the set of π-calculus names, $\mathrm{A}\pi_{\mathcal{P}}$ is the set of π-calculus processes, and $(m \leftrightarrow n)P$ denotes the process $\{m\leftrightarrow n\}\cdot P$ obtained by swapping the names m, n in the process P. Consider given a set $\mathcal{V}$ of names variables and a set χ of process variables.

Definition 2.2 (Term π-algebra) *The term π-algebra is the free π-algebra*

$$\mathbf{P} = \langle \mathcal{N}, \mathcal{I}, \mathbf{0}, |, \boldsymbol{\nu}, (\leftrightarrow)_{\mathcal{N}}, (\leftrightarrow)_{\mathcal{I}} \rangle$$

over a set of variables $x \in \mathcal{V}$ for the sort of name terms $\mathcal{N}$*, and over a set of variables $\mathcal{X} \in \chi$ for the sort of* indexes $\mathcal{I}$*. We use the meta-variables $m, n, p \in \mathcal{N}$; $u, v, t \in \mathcal{I}$; $\mathcal{X}, \mathcal{Y}, \mathcal{Z} \in \chi$; $x, y, z \in \mathcal{V}$; $\gamma \in \mathcal{G} = \mathcal{I} \cup \mathcal{N}$.*

The elements of the term π-algebra to which we have called *indexes* denote elements of the intended process algebra (processes, the worlds of our modal logic), while the *name terms* denote the pure names used in processes. Note that in the term π-algebra $(m \leftrightarrow n)P$ (resp. $(m \leftrightarrow n)p$) is a special process term (resp. name term) in which transposition is interpreted formally (*e.g.*, as an explicit transposition).

Definition 2.3 (Algebraic free variables) *Given an index or name term γ, we denote by afv(γ) its set of algebraic free (name and process) variables, defined simply as the collection of all the variables in $\mathcal{V}$ and χ occurring in such terms.*

Therefore a variable x is algebraic free, e.g., in the index $(\boldsymbol{\nu} x)\mathbf{0}$, while the name n is not free in the usual sense in the π-calculus process $(\boldsymbol{\nu} n)\mathbf{0}$. The notation $\tau\gamma$ will be used to represent the application of a transposition to some γ, and $\rho\gamma$ to represent the application of an arbitrary sequence of transpositions (a permutation) to γ.

Definition 2.4 (Interpretation) *Given any π-algebra Π, an* interpretation I of the term π-algebra into Π *is a pair of mappings $I_{\mathcal{L}} : \mathcal{V} \to \Pi_{\mathcal{L}}$ and $I_{\mathcal{P}} : \chi \to \Pi_{\mathcal{P}}$.*

Every interpretation I extends to the unique homomorphism $\hat{I} : \mathbf{P} \to \Pi$ of π-algebras in the standard way. Usually, we will write just σ for the homomorphic extension $\hat{\sigma}$ of a π-substitution σ.

Definition 2.5 (π-substitution) *A π-substitution is an interpretation into* **P**.

Every π-substitution σ extends to the homomorphism $\hat{\sigma} : \mathbf{P} \to \mathbf{P}$ of term π-algebras that acts as a syntactic substitution on indexes. We denote by $\{x \leftarrow n\}$ the π-substitution that maps x into n and behaves like the identity elsewhere, and likewise for $\{X \leftarrow u\}$. If $I_{\mathcal{L}}$ is a mapping $\mathcal{V} \to \mathcal{N}$ then we note by $I_{\mathcal{L}}\{x \leftarrow n\}$ the mapping $I'_{\mathcal{L}}$ such that $I'_{\mathcal{L}}(z) \triangleq I(z)$ for $z \neq x$ and $I'_{\mathcal{L}}(x) \triangleq n$. Likewise, if I is an interpretation, we write $I\{x \leftarrow n\}\{X \leftarrow u\}$ for the interpretation that behaves like I except that it maps x to n and X to u.

2.2 Constraint theories

The worlds of our logic relate to each other both by spatial and temporal constraints: spatial constraints express that the processes denoted by the equated indexes have the same spatial structure (*cf.* π-calculus structural congruence), while temporal constraints express that a process has a reduction to another process (*cf.* π-calculus reduction). A constraint theory is then the presentation of a particular world structure.

Definition 2.6 (Constraint and constraint theory) *A* constraint c *is either an index equation, a reduction, a name apartness, or a set apartness, defined by*

c ::=		Constraints	
	$u \doteq v$	Index equation	$(u, v \in \mathcal{I})$
	$u \to v$	Reduction	$(u, v \in \mathcal{I})$
	$m \# n$	Name apartness	$(m, n \in \mathcal{N}$
	$m \# X$	Set apartness	$(m \in \mathcal{N}, X \in \mathcal{X})$

A constraint theory *is a finite set of constraints.*

An equation $u \doteq v$ states that the indexes u and v denote structurally congruent processes, while a reduction $u \to v$ asserts that the process denoted by the index u reduces to the process denoted by the index v. In order to handle freshness constraints explicitly, we also introduce apartness constraints: $m \# n$ meaning that the name terms m and n denote distinct names, and $m \# X$ meaning that the name term m denotes a name distinct from any name in the (finite) support [2] of the property denoted by the propositional variable X (see Section 3.1).

Definition 2.7 (Closure of a constraint theory) *Given a constraint theory S, the relations $\doteq_S \subseteq \mathcal{N} \times \mathcal{N}$, $\#_S \subseteq \mathcal{N} \times \mathcal{N}$, $\#_S \subseteq \mathcal{N} \times \mathcal{X}$, $\doteq_S \subseteq \mathcal{I} \times \mathcal{I}$ and $\to_S \subseteq \mathcal{I} \times \mathcal{I}$ are inductively defined by the set of closure rules in Figs. 2-3.*

(Basic)

$u \doteq v \in S \Rightarrow u \doteq_S v$	(Basic Equ)
$m \# n \in S \Rightarrow m \#_S n$	(Basic Apart)
$u \rightarrow v \in S \Rightarrow u \rightarrow_S v$	(Basic Red)

(Spatial)

$u\|\mathbf{0} \doteq_S u$	(Sp Void)
$u\|v \doteq_S v\|u$	(Sp Par Comm)
$(u\|v)\|t \doteq_S u\|(v\|t)$	(Sp Par Assoc)
$(\boldsymbol{\nu} n)\mathbf{0} \doteq_S \mathbf{0}$	(Sp Res Void)
$(\boldsymbol{\nu} n)(\boldsymbol{\nu} n)u \doteq_S (\boldsymbol{\nu} n)u$	(Sp Res Res)
$(\boldsymbol{\nu} m)(\boldsymbol{\nu} n)u \doteq_S (\boldsymbol{\nu} n)(\boldsymbol{\nu} m)u$	(Sp Res Comm)
$(\boldsymbol{\nu} n)(u\|(\boldsymbol{\nu} n)v) \doteq_S ((\boldsymbol{\nu} n)u)\|(\boldsymbol{\nu} n)v$	(Sp Res Par)

(Congruence)

$\gamma \doteq_S \gamma$	(Cong Refl)
$\gamma \doteq_S \gamma' \Rightarrow \gamma' \doteq_S \gamma$	(Cong Sym)
$\gamma \doteq_S \gamma', \gamma' \doteq_S \gamma'' \Rightarrow \gamma \doteq_S \gamma''$	(Cong Trans)
$u \doteq_S v \Rightarrow u\|t \doteq_S v\|t$	(Cong Par)
$u \doteq_S v, m \doteq_S n \Rightarrow (\boldsymbol{\nu} m)u \doteq_S (\boldsymbol{\nu} n)v$	(Cong Res)
$m \doteq_S n, \gamma \doteq_S \gamma' \Rightarrow (m \leftrightarrow r)\gamma \doteq_S (r \leftrightarrow n)\gamma'$	(Cong Swap)

Figure 2: Closure of constraint theories.

Closure rules axiomatize basic structural properties of our intended models. For instance, (Spatial) characterizes the basic properties of structural congruence; in particular (Sp Res Par) expresses the name extrusion property of π-calculus.

If c is a constraint and S is a constraint theory, we will sometimes write $S \vdash c$ to state that c holds in the closure of S, writing $S \vdash \gamma \doteq \gamma'$ for $\gamma \doteq_S \gamma'$, and likewise for $\#_S$ and $\rightarrow_S$. N.B. We use the symbol $\doteq_S$ both to denote equality between name terms and indexes under S, and similarly for $\#_S$, used both in $m \#_S n$ and $m \#_S X$.

Lemma 2.8 *We have*

1. $S \vdash c$ *implies* $S \cup S' \vdash c$.
2. *If* $S \vdash c$ *and* $S, c \vdash c'$ *then* $S \vdash c'$.
3. *If* $S \vdash c$ *then* $\sigma(S) \vdash \sigma(c)$.

An interpretation for a constraint theory assigns an appropriate denotation to all propositional, process and name variables occurring on it. As in Part I, we focus on a version of the spatial logic for the asynchronous π-calculus (we use the standard notations $\equiv$ and $\rightarrow$ for asynchronous π-calculus structural congruence and reduction). Hence, interpretations will map process variables into Aπ processes, name variables into Aπ names (this part is just a π-algebra interpretation into Aπ), and propositional variables into property sets (see Section 1.2).

Definition 2.9 (Logical interpretation into Aπ**)** *A logical interpretation* $\mathcal{J}$ *into* Aπ *is a pair* (v, I) *where* v *is a valuation and* I *is an interpretation into* Aπ.

(Apartness)

$m \#_S p, n \#_S p \Rightarrow (m \leftrightarrow n)p \doteq_S p$	(Swap Fresh)
$m \#_S n \Rightarrow n \#_S m$	(Apart Sym)
$p \#_S q, p \doteq_S p', q \doteq_S q' \Rightarrow p' \#_S q'$	(Cong Apr Name)
$p \#_S X, p \doteq_S q \Rightarrow q \#_S X$	(Cong Apr Set)

(Transposition)

$\tau \mathbf{0} \doteq_S \mathbf{0}$	(Swap Void)
$\tau(u \mid v) \doteq_S \tau u \mid \tau v$	(Swap Par)
$\tau(\boldsymbol{\nu} p)u \doteq_S (\boldsymbol{\nu} \tau p)\tau u$	(Swap Res)
$\tau(p \leftrightarrow q)\gamma \doteq_S (\tau p \leftrightarrow \tau q)\tau\gamma$	(Swap Swap)
$\tau\tau\gamma \doteq_S \gamma$	(Swap Inv)
$(n \leftrightarrow n)\gamma \doteq_S \gamma$	(Swap Id)
$(m \leftrightarrow n)m \doteq_S n$	(Swap App)
$u \doteq_S (\boldsymbol{\nu} n)t, u \doteq_S (\boldsymbol{\nu} m)v \Rightarrow (n \leftrightarrow m)u \doteq_S u$	(Swap Erase)

(Reduction)

$u \rightarrow_S t, v \doteq_S u, t \doteq_S w \Rightarrow v \rightarrow_S w$	(Red Cong)
$u \rightarrow_S t \Rightarrow u \mid v \rightarrow_S t \mid v$	(Red Par)
$u \rightarrow_S t \Rightarrow (\boldsymbol{\nu} n)u \rightarrow_S (\boldsymbol{\nu} n)t$	(Red Res)
$u \rightarrow_S t \Rightarrow \tau u \rightarrow_S \tau t$	(Red Transp)

Figure 3: Closure of constraint theories.

Definition 2.10 (Satisfaction and Validity) *The relation of* satisfaction *between a logical interpretation $\mathcal{J} = (v, I)$ into* Aπ *and constraints is defined thus:*

$$\begin{array}{llllll} \mathcal{J} \textit{ satisfies } u \doteq v & \Leftrightarrow & I_{\mathcal{P}}(u) \equiv I_{\mathcal{P}}(v) & \mathcal{J} \textit{ satisfies } u \rightarrow v & \Leftrightarrow & I_{\mathcal{L}}(u) \rightarrow I_{\mathcal{L}}(v) \\ \mathcal{J} \textit{ satisfies } n \# m & \Leftrightarrow & I_{\mathcal{L}}(n) \neq I_{\mathcal{L}}(m) & \mathcal{J} \textit{ satisfies } n \# X & \Leftrightarrow & I_{\mathcal{L}}(n) \notin \textit{supp}(v(X)) \end{array}$$

$\mathcal{J}$ satisfies *the constraint theory S if $\mathcal{J}$ satisfies all constraints in S. A constraint $S \vdash c$* is valid *if every interpretation that satisfies S also satisfies c.*

The following establishes soundness of the closure of constraint theories with respect to the intended asynchronous π-calculus (Aπ) model in [2]. We have

Lemma 2.11 (Soundness for Aπ**)** *Let S be a constraint theory and $\mathcal{J} = (v, I)$ a logical interpretation into* Aπ *that satisfies S. We have:*

1. *$u \doteq_S t$ implies $I_{\mathcal{P}}(u) \equiv I_{\mathcal{P}}(t)$ and $m \doteq_S n$ implies $I_{\mathcal{L}}(m) = I_{\mathcal{L}}(n)$.*
2. *If $u \rightarrow_S t$ then $I_{\mathcal{P}}(u) \rightarrow I_{\mathcal{P}}(t)$.*
3. *$m \#_S n$ implies $I_{\mathcal{L}}(m) \neq I_{\mathcal{L}}(n)$ and $m \#_S X$ implies $I_{\mathcal{L}}(m) \notin \textit{supp}(v(X))$.*

Proof. By induction on the derivations of $\gamma \doteq_S \gamma'$, $n \#_S m$ and $u \rightarrow_S v$ using well-known properties of structural congruence, name transposition and reduction of the asynchronous π-calculus Aπ. ■

3 Sequent Calculus

The proof system for our logic is presented along the lines of a quite standard Gentzen-style sequent calculus.

m, n, p ::=		Name Terms	$(m, n, p \in \mathcal{N})$
	x	Name variable	$(x \in \mathcal{V})$
	$(m \leftrightarrow n)p$	Transposition term	
A, B ::=		Formulas	$(A, B \in \Phi)$
	$\mathtt{F}$	False	
	$A \wedge B$	Conjunction	
	$A \Rightarrow B$	Implication	
	$\mathbf{0}$	Void	
	$A \vert B$	Composition	
	$A \triangleright B$	Guarantee	
	$m®A$	Revelation	
	$m \oslash A$	Hiding	
	$m\langle n \rangle$	Message	
	$\Diamond A$	Next	
	$\forall x.A$	First-order universal quantification	
	$Ӈx.A$	Freshness quantification	
	X	Propositional variable	$(X \in \mathcal{X})$
	$\forall X.A$	Second-order universal quantification	

Formulas are subject to: no occurrence of a variable x in a transposition term is under the scope of a $\forall x.$ or a $Ӈx$.

Figure 4: Formulas

3.1 Formulas

Formulas, defined in Figure 4, include classical propositional connectives, $\mathtt{F}$, $\wedge$, $\Rightarrow$, and the basic spatial operators: $A|B$ (the tensor, representing the parallel composition of processes), $\mathbf{0}$ (the unit of the tensor, representing the collection of void processes), and $A \triangleright B$ (the linear implication associated with the tensor). This last operator corresponds to context-system specification of processes, which are the concurrency-theory equivalent of pre/post conditions. Name hiding induces a pair of adjunct logical operators. The formula $n®A$ means that a hidden name, which we choose to call n, exists in a restricted scope that satisfies property A. It is matched by a π-calculus term $(\boldsymbol{\nu} n)u$ provided that u satisfies A (see rule (®R) in Figure 6, and the example in Section 5.5; see [5, 2] for further details.) The formula $A \oslash n$ is the logical adjunct of $n®A$. The notion of *fresh name* is introduced by a quantifier $Ӈx.A$; this means that x denotes a name that is fresh with respect to the names used either in A or in processes satisfying A. $Ӈx.A$ is defined along the lines of the freshness quantifier of Gabbay-Pitts [7, 11], and its semantics is designed to be compatible with recursive formulas. A logical operator $n\langle m \rangle$ allows us to assert that a message m is present over channel n, giving us some minimal power to observe the behavior of processes. A next-step temporal operator, $\Diamond A$, allows us to talk about the behavior of a process after a single (unspecified) reduction step. Finally, we have a second-order quantifier and related propositional variables.

Formulas are the same as in Part I [2], except that (1) recursion is now replaced by second-order universal quantification, and (2) we allow formulas to contain general name terms (including explicit transpositions), and not only variables and pure names. To avoid any confusion, we will refer to the formulas as defined in [2] by *simple* formulas.

In $\forall x.A$, $Ӈx.A$ (and $\forall X.A$), the variables x (and X) are bound with scope the formula A. We assume defined on formulas the standard relation $\equiv_\alpha$ of α-conversion (safe renaming of bound variables), but will never implicitly take formulas "up to α-conversion": our manipulation of variables via α-conversion steps will always be quite explicit. We also assume defined as usual the set *fv*(A) of *free name variables* in A, and the set *fpv*(A) of *free propositional variables* in A. Then, we define the set of *logically free variables* of a formula A by *lfv*$(A) \triangleq$ *fv*$(A) \cup$ *fpv*(A). If m is a name term and A is a formula then $A\{x \leftarrow m\}$ denotes the formula obtained by replacing of all free occurrences of x in A by the name term m, (nondeterministically) renaming bound variables as needed to avoid capture of names in m.

Any substitution $I_{\mathcal{L}} : \mathcal{V} \to \Lambda$ of pure names for name variables acts on a formula A (respectively, on a name term p) yielding a simple formula $I_{\mathcal{L}}(A)$ (respectively a pure name), defined as expected. The following definition lifts the equational theory of name terms of Definition 2.7 to the level of formulas: $A \equiv_S B$ asserts that, under any substitution that satisfies all constraints in S, A and B denote the same simple formula.

Definition 3.1 (Equational equivalence of formulas) *Given a constraint theory S, $\equiv_S$ is the least equivalence relation on formulas containing all pairs $(A\{x \leftarrow m\}, A\{x \leftarrow n\})$ for all name terms m, n such that $m \doteq_S n$.*

The semantics of the freshness quantifier requires names to be chosen fresh with respect to the free names of both processes and formulas. By the formation condition at the bottom of Figure 4, transposition terms in formulas *never contain bound occurrences of name variables*. Hence, a maximal term in a formula is either a bound name variable, or a term completely built out of free name variables: explicit transpositions in formulas just swap *free names*. Thus, it is sensible to define the set *ft*(A) of *free terms* of a formula A to be the set of maximal name terms occurring in A which are not a bound occurrence of a variable. We then introduce

Definition 3.2 (Freshness of name terms wrt. formulas) *Given a constraint theory S and a formula A, we write $n \#_S A$ as an abbreviation of $n \#_S$ ft(A).*

Thus, $n \#_S A$ asserts that, under the constraints in S, n denotes a name *distinct* from all names denoted by the (free) terms in formula A. We can verify that the relations $\equiv_S$ (between formulas) and $\#_S$ (between name terms and formulas) defined in Definition 3.1 and Definition 3.2 are sound with respect to their intended interpretation.

3.2 Sequents

A context (Γ, Δ) is a finite multiset of indexed formulas of the form $u : A$ where u is an index and A is a formula.

Definition 3.3 (Sequent) *A* sequent *is a judgment $\langle S \rangle\, \Gamma \vdash \Delta$ where S is a constraint theory, and Δ and Γ are logical contexts.*

The *right context* Δ is interpreted as the disjunction of its formulas, the *left context* Γ is interpreted as the conjunction of the formulas in it. Defining contexts as multisets allows for the implicit use of exchange in proofs. We write $\Delta \equiv_\alpha \Delta'$ if Δ' is obtained from Δ by α-converting some formulas in it. We have

Definition 3.4 (Variables in sequents) *The set of free (name, process, and propositional) variables of a context Δ is given by lfv$(\Delta) = \bigcup\{$afv$(u) \cup$ lfv$(A) \mid u : A \in \Delta\}$. The set of free (name, process, and propositional) variables in a sequent $\langle S \rangle\, \Gamma \vdash \Delta$ is given by afv$(S) \cup$ fv$(\Gamma) \cup$ fv(Δ).*

N.B.: name variables x occur both in indexes u and in formulas A; process variables $\mathcal{X}$ occur only in indexes; propositional variables X occur only in formulas. Semantics of formulas is defined in Part I by a mapping $[\![-]\!]_v$ that assigns a property set $[\![A]\!]_v$ to any simple formula A, given a valuation v. Given a logical interpretation $\mathcal{J} = (v, I)$ and a context Γ, we say that $\mathcal{J}$ *satisfies all of* Γ if $I(u) \in [\![I_{\mathcal{L}}(A)]\!]_v$ for all $u : A \in \Gamma$. Likewise, we say that $\mathcal{J}$ *satisfies some of* Γ if $I(u) \in [\![I_{\mathcal{L}}(A)]\!]_v$ for some $u : A \in \Gamma$. Hence we have

Definition 3.5 (Valid Sequent in $\mathrm{A}\pi$**)** *A sequent* $\langle S \rangle\, \Gamma \vdash \Delta$ *is valid in* $\mathrm{A}\pi$ *if for all interpretations* $\mathcal{J}$ *into* $\mathrm{A}\pi$ *such that* $\mathcal{J}$ *satisfies all constraints in* S*, and* $\mathcal{J}$ *satisfies all of* Γ*, then* $\mathcal{J}$ *satisfies some of* Δ.

3.3 Inference Rules

Inference rules may have for premises both sequents $\langle S \rangle\, \Gamma \vdash \Delta$ and assertions over the closure of the theory S of the form $u \doteq_S v$ (mostly in the rules for spatial connectives), $A \equiv_S B$ (in the identity axiom), $u \rightarrow_S v$ (in the temporal rules) or $n \,\#_S\, A$ (in the freshness rules).

The rules in the identity, structural and propositional groups (see Figures 5) follow the standard. Note that in (Id) indexes are identified up to $\doteq_S$, while formulas are identified up to $\equiv_S$: (Id) absorbs the simple theory of equality captured by $\equiv_S$, thus axiomatizing the principle of substitution of equals for equals of name terms in formulas. We include explicit contraction rules (CL) and (CR); weakening is admissible, and exchange may be dealt with implicitly, since sequent contexts are multisets.

In the rules for propositional connectives, indexes keep track of the processes for which the formulas are asserted to hold, but do not interfere in any way with the constraint part of sequents.

The rules for the spatial connectives (Figure 6) make essential use of the constraint theories in sequents. Note that the left rules, when read bottom-up, introduce spatial constraints into the constraint theories, and the respective right rules, when read top-down, check corresponding constraints. While spatial rules rely on spatial constraints, temporal rules (Figure 7) rely on reduction constraints.

The rules for first and second order quantifiers have the expected form (Figure 8). We then introduce the rules for freshness (Figure 9). Rule (И) asserts, when read bottom-up, that there is always a name (denoted by) x that is fresh with respect to the free names of (the process denoted by) the index u, and that is also fresh with respect to a set of names (denoted by the name and propositional variables in) N. Hence, rule (И) corresponds to the (Fresh) axiom of Pitts' Nominal Logic [11]. The transposition rule (τ) captures the property of invariance of the semantics under transposition of names (see the main theorem of Part I [2]). Moreover, as we shall discuss in Section 4.3, explicit transpositions and the transposition rule play an crucial role in obtaining cut-elimination for the freshness quantifier. By $\{m \leftrightarrow n\} \cdot A$ we denote the formula obtained by swapping occurrences of m and n in A, possibly by introducing explicit transpositions: if $A[n_1, \ldots, n_k]$ is a formula with free terms $n_1, \ldots, n_k$, then $\tau \cdot A[n_1, \ldots, n_k]$ is the formula $A[\tau \cdot n_1, \ldots, \tau \cdot n_k]$.

The rules (ИL/R) for the fresh quantifier do not show the symmetry one might expect of a left / right rule pair. This fact relates to the existential / universal ambivalence of freshness quantification (the Gabbay-Pitts property): note that (ИL) follows the pattern of ($\forall$L), while (ИR) follows the pattern of ($\exists$R). Then, (И) embodies the introduction of fresh witnesses usually present in *both* ($\forall$R) and ($\exists$L). Besides the freshness condition $n \,\#_S\, \text{И}x.A$ of the name denoted by n with respect to the free names in the formula $\text{И}x.A$, the assumption $u \doteq_S (\boldsymbol{\nu} n)v$ ensures that n denotes a name that does not occur free in the process denoted by u, *cf.* the semantics of $\text{И}x.A$.

As discussed in Section 1.3, world rules (Figure 10) axiomatize certain deep (extra-logical) properties of the worlds (inversion principles for structural congruence and process reduction). Note that none of the studied proof-theoretic properties of our logic (e.g., cut-elimination) depend on the chosen set of world rules, provided they do not change the logical contexts of sequents, and just check and eliminate constraints from the constraint part of sequents in their premises.

We assert $\vdash \langle S \rangle\, \Gamma \vdash \Delta$ to state that the sequent $\langle S \rangle\, \Gamma \vdash \Delta$ has a derivation. We can show that all rules are sound with respect to the intended model of Part I [2]:

Theorem 3.6 (Soundness) *All derivable sequents are valid in* $\mathrm{A}\pi$.

$$\frac{u \doteq_S u' \quad A \equiv_S A'}{\langle S\rangle\, \Gamma, u : A \vdash u' : A', \Delta}\ (\mathrm{Id}) \qquad \frac{\langle S\rangle\, \Gamma \vdash u : A, \Delta \quad \langle S\rangle\, \Gamma, u : A \vdash \Delta}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\mathrm{Cut})$$

$$\frac{\langle S\rangle\, \Gamma, u : A, u : A \vdash \Delta}{\langle S\rangle\, \Gamma, u : A \vdash \Delta}\ (\mathrm{CL}) \qquad \frac{\langle S\rangle\, \Gamma \vdash u : A, u : A, \Delta}{\langle S\rangle\, \Gamma \vdash u : A, \Delta}\ (\mathrm{CR})$$

$$\frac{}{\langle S\rangle\, \Gamma, u : \mathtt{F} \vdash \Delta}\ (\mathtt{F}\mathrm{L}) \qquad \frac{\langle S\rangle\, \Gamma \vdash \Delta}{\langle S\rangle\, \Gamma \vdash u : \mathtt{F}, \Delta}\ (\mathtt{F}\mathrm{R})$$

$$\frac{\langle S\rangle\, \Gamma, u : A, u : B \vdash \Delta}{\langle S\rangle\, \Gamma, u : A \wedge B \vdash \Delta}\ (\wedge\mathrm{L}) \qquad \frac{\begin{array}{c}\langle S\rangle\, \Gamma \vdash u : A, \Delta \\ \langle S\rangle\, \Gamma \vdash u : B, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u : A \wedge B, \Delta}\ (\wedge\mathrm{R})$$

$$\frac{\langle S\rangle\, \Gamma \vdash u : A, \Delta \quad \langle S\rangle\, \Gamma, u : B \vdash \Delta}{\langle S\rangle\, \Gamma, u : A \Rightarrow B \vdash \Delta}\ (\Rightarrow\mathrm{L}) \qquad \frac{\langle S\rangle\, \Gamma, u : A \vdash u : B, \Delta}{\langle S\rangle\, \Gamma \vdash u : A \Rightarrow B, \Delta}\ (\Rightarrow\mathrm{R})$$

Figure 5: Propositional Rules.

$$\frac{\langle S, t \doteq \mathbf{0}\rangle\, \Gamma \vdash \Delta}{\langle S\rangle\, \Gamma, t : \mathbf{0} \vdash \Delta}\ (\mathbf{0}\mathrm{L}) \qquad \frac{u \doteq_S \mathbf{0}}{\langle S\rangle\, \Gamma \vdash u : \mathbf{0}, \Delta}\ (\mathbf{0}\mathrm{R})$$

$$\frac{\begin{array}{c}[\mathcal{X} \textit{ and } \mathcal{Y} \textit{ not free in the conclusion}] \\ \langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : A, \mathcal{Y} : B \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : A|B \vdash \Delta}\ (|\mathrm{L}) \qquad \frac{\begin{array}{c}\langle S\rangle\, \Gamma \vdash v : A, \Delta \\ \langle S\rangle\, \Gamma \vdash t : B, \Delta \quad u \doteq_S v|t\end{array}}{\langle S\rangle\, \Gamma \vdash u : A|B, \Delta}\ (|\mathrm{R})$$

$$\frac{\langle S\rangle\, \Gamma \vdash t : A, \Delta \quad \langle S\rangle\, \Gamma, t|u : B \vdash \Delta}{\langle S\rangle\, \Gamma, u : A \triangleright B \vdash \Delta}\ (\triangleright\mathrm{L}) \qquad \frac{\begin{array}{c}[\mathcal{X} \textit{ not free in the conclusion}] \\ \langle S\rangle\, \Gamma, \mathcal{X} : A \vdash v : B, \Delta \quad v \doteq_S \mathcal{X}|u\end{array}}{\langle S\rangle\, \Gamma \vdash u : A \triangleright B, \Delta}\ (\triangleright\mathrm{R})$$

$$\frac{\begin{array}{c}[\mathcal{X} \textit{ not free in the conclusion}] \\ \langle S, u \doteq (\boldsymbol{\nu} n)\mathcal{X}\rangle\, \Gamma, \mathcal{X} : A \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : n\circledR A \vdash \Delta}\ (\circledR\mathrm{L}) \qquad \frac{\langle S\rangle\, \Gamma \vdash u : A, \Delta \quad t \doteq_S (\boldsymbol{\nu} n)u}{\langle S\rangle\, \Gamma \vdash t : n\circledR A, \Delta}\ (\circledR\mathrm{R})$$

$$\frac{\langle S\rangle\, \Gamma, t : A \vdash \Delta \quad t \doteq_S (\boldsymbol{\nu} n)u}{\langle S\rangle\, \Gamma, u : A\oslash n \vdash \Delta}\ (\oslash\mathrm{L}) \qquad \frac{\langle S\rangle\, \Gamma \vdash u : A, \Delta \quad u \doteq_S (\boldsymbol{\nu} n)t}{\langle S\rangle\, \Gamma \vdash t : A\oslash n, \Delta}\ (\oslash\mathrm{R})$$

Figure 6: Spatial Rules.

$$\frac{\begin{array}{c}[\mathcal{X} \textit{ not free in the conclusion}] \\ \langle S, u \to \mathcal{X}\rangle\, \Gamma, \mathcal{X} : A \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : \Diamond A \vdash \Delta}\ (\Diamond\mathrm{L}) \qquad \frac{\langle S\rangle\, \Gamma \vdash v : A, \Delta \quad u \to_S v}{\langle S\rangle\, \Gamma \vdash u : \Diamond A, \Delta}\ (\Diamond\mathrm{R})$$

Figure 7: Temporal Rules.

$$\frac{\langle S\rangle\, \Gamma, u : A\{x{\leftarrow}n\} \vdash \Delta}{\langle S\rangle\, \Gamma, u : \forall x.A \vdash \Delta}\ (\forall\mathrm{L}) \qquad \frac{\begin{array}{c}[y \textit{ not free in the conclusion}]\\ \langle S\rangle\, \Gamma \vdash u : A\{x{\leftarrow}y\}, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u : \forall x.A, \Delta}\ (\forall\mathrm{R})$$

$$\frac{\langle S\rangle\, \Gamma, u : A\{X{\leftarrow}B\} \vdash \Delta}{\langle S\rangle\, \Gamma, u : \forall X.A \vdash \Delta}\ (\forall^2\mathrm{L}) \qquad \frac{\begin{array}{c}[Y \textit{ not free in the conclusion}]\\ \langle S\rangle\, \Gamma \vdash u : A\{X{\leftarrow}Y\}, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u : \forall X.A, \Delta}\ (\forall^2\mathrm{R})$$

(In ($\forall$L) and ($\forall$R), the formula $\forall x.A$ must verify the condition in Fig. 4)

Figure 8: Quantifier Rules.

$$\frac{\begin{array}{c}[\mathcal{X}, x \textit{ not free in the conclusion, } u \textit{ or } N]\\ \langle S, x \,\#\, N, u \doteq (\boldsymbol{\nu} x)\mathcal{X}\rangle\, \Gamma \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\text{И}) \qquad \frac{\begin{array}{c}m, n \,\#_S\, \mathit{fpv}(A)\\ \langle S\rangle\, \Gamma, (m\leftrightarrow n)u : \{m\leftrightarrow n\}\cdot A \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : A \vdash \Delta}\ (\tau)$$

$$\frac{\begin{array}{c}u \doteq_S (\boldsymbol{\nu} n)v\\ n \,\#_S\, \text{И}x.A\\ \langle S\rangle\, \Gamma, u : A\{x{\leftarrow}n\} \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : \text{И}x.A \vdash \Delta}\ (\text{И}\mathrm{L}) \qquad \frac{\begin{array}{c}u \doteq_S (\boldsymbol{\nu} n)v\\ n \,\#_S\, \text{И}x.A\\ \langle S\rangle\, \Gamma \vdash u : A\{x{\leftarrow}n\}, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u : \text{И}x.A, \Delta}\ (\text{И}\mathrm{R})$$

(In (ИL) and (ИR), the formula $\text{И}x.A$ must verify the condition in Fig. 4)

Figure 9: Freshness Rules

$$\frac{\langle S, u \doteq \mathbf{0}\rangle\, \Gamma \vdash \Delta \quad u|v \doteq_S \mathbf{0}}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\mathrm{S}|\mathbf{0}) \qquad \frac{\langle S, u \doteq \mathbf{0}\rangle\, \Gamma \vdash \Delta \quad (\boldsymbol{\nu} n)u \doteq_S \mathbf{0}}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\mathrm{S}\boldsymbol{\nu}\mathbf{0})$$

$$\frac{\begin{array}{c}[\mathcal{X} \textit{ and } \mathcal{Y} \textit{ not free in the conclusion}]\\ \langle S, u \doteq \mathcal{X}|\mathcal{Y}, (\boldsymbol{\nu} n)\mathcal{X} \doteq t, (\boldsymbol{\nu} n)\mathcal{Y} \doteq v\rangle\, \Gamma \vdash \Delta \quad (\boldsymbol{\nu} n)u \doteq_S t|v\end{array}}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\mathrm{S}\boldsymbol{\nu}|)$$

$$\frac{\begin{array}{c}[\mathcal{X}, \mathcal{X}', \mathcal{Y} \textit{ and } \mathcal{Y}' \textit{ not free in the conclusion}]\\ \langle S, u \doteq \mathcal{X}|\mathcal{X}', w \doteq \mathcal{Y}|\mathcal{Y}', t \doteq \mathcal{X}|\mathcal{Y}, v \doteq \mathcal{X}'|\mathcal{Y}'\rangle\, \Gamma \vdash \Delta \quad u|w \doteq_S t|v\end{array}}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\mathrm{S}||)$$

$$\frac{\begin{array}{c}[\mathcal{X} \textit{ not free in the conclusion}]\\ \langle S, u \doteq (m\leftrightarrow n)v\rangle\, \Gamma \vdash \Delta\\ \langle S, u \doteq (\boldsymbol{\nu} m)\mathcal{X}, v \doteq (\boldsymbol{\nu} n)\mathcal{X}\rangle\, \Gamma \vdash \Delta \quad (\boldsymbol{\nu} n)u \doteq_S (\boldsymbol{\nu} m)v\end{array}}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\mathrm{S}\boldsymbol{\nu}\boldsymbol{\nu})$$

$$\frac{\mathbf{0} \rightarrow_S u}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\mathrm{S}\mathbf{0}\rightarrow) \qquad \frac{\begin{array}{c}[\mathcal{X} \textit{ not free in the conclusion}]\\ \langle S, u \rightarrow \mathcal{X}, v \doteq (\boldsymbol{\nu} n)\mathcal{X}\rangle\, \Gamma \vdash \Delta \quad (\boldsymbol{\nu} n)u \rightarrow_S v\end{array}}{\langle S\rangle\, \Gamma \vdash \Delta}\ (\mathrm{S}\boldsymbol{\nu}\rightarrow)$$

Figure 10: World Rules.

4 Basic Proof Theory

In this Section we develop some proof-theory for our logic, stating several admissible proof principles and a cut elimination result for the first-order fragment.

4.1 Admissible Rules

Most of the presented proof principles are size-preserving, and instrumental to the proof of cut elimination. We introduce a measure for the *size* of a derivation, in which certain occurrences of the (τ) rule are not weighted. An occurrence of the inference rule (τ) in a derivation is *simple* if it applies either to an instance of (Id) or to another simple occurrence of (τ). Then

Definition 4.1 (Size of a derivation) *The* size of a derivation *is the number of rule occurrences it contains, other than simple occurrences of rule (τ).*

We then assert $\vdash_n \langle S \rangle\, \Gamma \vdash \Delta$ to state that the given sequent has a derivation of size not exceeding n. We have the following useful admissible rules

Lemma 4.2 (Basic) *The following proof principles are admissible:*

$$\frac{\begin{array}{c}[\ \varphi' \textit{ is not free in the premise }]\\ \vdash_n \langle S \rangle\, \Gamma \vdash \Delta\end{array}}{\vdash_n \langle S\{\varphi \leftarrow \varphi'\} \rangle\, \Gamma\{\varphi \leftarrow \varphi'\} \vdash \Delta\{\varphi \leftarrow \varphi'\}}\ \text{(Ren)} \qquad \frac{\begin{array}{c}[\ \Gamma \equiv_\alpha \Gamma' \textit{ and } \Delta \equiv_\alpha \Delta'\ ]\\ \vdash_n \langle S \rangle\, \Gamma \vdash \Delta\end{array}}{\vdash_n \langle S \rangle\, \Gamma' \vdash \Delta'}\ (\alpha)$$

$$\frac{\vdash_n \langle S \rangle\, \Gamma \vdash \Delta}{\vdash_n \langle S, S' \rangle\, \Gamma, \Gamma' \vdash \Delta, \Delta'}\ \text{(W)} \qquad \frac{\vdash_n \langle S, c \rangle\, \Gamma \vdash \Delta \quad S \vdash c}{\vdash_n \langle S \rangle\, \Gamma \vdash \Delta}\ \text{(CS)}$$

$$\frac{\vdash_n \langle S \rangle\, \Gamma \vdash \Delta}{\vdash_n \langle S\{\mathcal{X} \leftarrow u\} \rangle\, \Gamma\{\mathcal{X} \leftarrow u\} \vdash \Delta\{\mathcal{X} \leftarrow u\}}\ (\text{Inst}\mathcal{I}) \qquad \frac{\vdash_n \langle S \rangle\, \Gamma \vdash \Delta}{\vdash_n \langle S\{x \leftarrow m\} \rangle\, \Gamma\{x \leftarrow m\} \vdash \Delta\{x \leftarrow m\}}\ (\text{Inst}\mathcal{N})$$

Lemma 4.3 (Replacement and Instantiation) *The following inference rules are admissible*

$$\frac{\begin{array}{c}[\ \mathcal{X} \textit{ not free in } S\ ]\\ \langle S \rangle\, \mathcal{X} : A \vdash \mathcal{X} : B \quad \langle S \rangle\, \mathcal{X} : B \vdash \mathcal{X} : A\end{array}}{\langle S \rangle\, \mathcal{Y} : C[A] \vdash \mathcal{Y} : C[B]}\ \text{(Repl)} \qquad \frac{\begin{array}{c}[\ X \textit{ not free in the conclusion }]\\ \langle S \rangle\, \Gamma \vdash \Delta\end{array}}{\langle S \rangle\, \Gamma\{X \leftarrow A\} \vdash \Delta\{X \leftarrow A\}}\ \text{(Inst2)}$$

All proofs for Lemmas 4.2 and 4.3 are carried out by induction on the size of derivations, except (Repl), which is by induction on the structure of the context C.

4.2 Derived Connectives and Inference Rules

We introduce some useful derived connectives (see Figure 11). These include the usual operations of the classical predicate calculus, namely $\neg A$ (Negation), $\exists x.A$ (Existential quantification), $A \vee B$ (Disjunction) and $\mathtt{T}$ (True), with the expected meaning. *Decomposition* $A \| B$ is the DeMorgan dual of composition $A|B$. For instance, a process satisfies $\mathbf{0}\|\mathbf{0}$ if it is single-threaded (or void). We also have the standard temporal modality $\Box$, the dual of $\Diamond$. The free name predicate ©n holds of any process with some free occurrence of the name (denoted by) n. *Inconsistency* $\sim A$ expresses internally to the logic that A is false of every process and *validity* $!A$ that A holds of every process [4]. Thus, *entailment* $A \mapsto B$ internalizes the consequence relation induced by the logic. The hidden name quantifier is defined as in [2]. For these connectives, the inference rules in Figure 12 can easily be shown to be admissible

$$
\begin{array}{llll}
\neg A & \triangleq & A \Rightarrow \mathtt{F} & \text{(Negation)} \\
\mathtt{T} & \triangleq & \neg \mathtt{F} & \text{(True)} \\
A \vee B & \triangleq & \neg A \Rightarrow B & \text{(Disjunction)} \\
A \| B & \triangleq & \neg(\neg A | \neg B) & \text{(Decomposition)} \\
©n & \triangleq & \neg n®\mathtt{T} & \text{(Free name)} \\
\Box A & \triangleq & \neg \Diamond \neg A & \text{(All next)} \\
\sim A & \triangleq & A \triangleright \mathtt{F} & \text{(Inconsistency)} \\
!A & \triangleq & \sim \neg A & \text{(Validity)} \\
A \mapsto B & \triangleq & !(A \Rightarrow B) & \text{(Entailment)} \\
\exists x.A & \triangleq & \neg \forall x. \neg A & \text{(First-order existential quantification)} \\
\exists X.A & \triangleq & \neg \forall X. \neg A & \text{(Second-order existential quantification)} \\
\mathsf{H} x.A & \triangleq & \text{И} x.x®A & \text{(Hidden name quantification)}
\end{array}
$$

Figure 11: Derived Connectives.

4.3 Cut Elimination

We show here that the first-order fragment of the spatial logic enjoys the cut elimination property (in this Section we restrict to first-order formulas and derivations, where a first-order derivation is a derivation without occurrences of second-order quantifiers). This result is reasonable evidence that our addition of structural and freshness constraints to sequents and inference rules is rather canonical. For instance, cuts on spatial formulas are eliminated in a rather uniform way, by matching process eigenvariables (on one side) against the given witnesses (on the other), and then eliminating the remaining (redundant) structural constraints. The cut case for freshness quantifications deserves a more detailed discussion. Consider then the following cut

$$
\frac{\dfrac{\langle S \rangle\, \Gamma \vdash u : A\{x{\leftarrow}n\}, \Delta}{\langle S \rangle\, \Gamma \vdash u : \text{И}x.A, \Delta} \qquad \dfrac{\langle S \rangle\, \Gamma, u : A\{x{\leftarrow}m\} \vdash u : \Delta}{\langle S \rangle\, \Gamma, u : \text{И}x.A \vdash \Delta}}{\langle S \rangle\, \Gamma \vdash \Delta}
$$

To eliminate this we need to cut $u : A\{x{\leftarrow}n\}$ against $u : A\{x{\leftarrow}m\}$, preserving the sequent contexts Γ, Δ untouched. In general m and n are name terms denoting distinct names (we could even have $m \#_S n$ provably). In fact, soundness of this cut follows from the fact that a fresh name is (in the sense of equivariance in nominal logic) equivalent to any other fresh name. The proof-theoretic counterpart of equivariance has as consequence that, in the given conditions, we can transform the derivation of $\langle S \rangle\, \Gamma \vdash u : A\{x{\leftarrow}m\}, \Delta$ into a derivation of $\langle S \rangle\, \Gamma \vdash u : A\{x{\leftarrow}n\}, \Delta$ verifying certain convenient properties, as stated in the following Lemma.

Lemma 4.4 (Closure) *Let ρ be a permutation of name terms $\mathfrak{n}$, each verifying $\mathfrak{n} \#_S fpv(A)$ and assume $u \doteq_S \rho t$ and $A' \equiv_S \rho{\cdot}A$. Then the following proof principles are admissible for the first-order fragment of the logic*

$$
\frac{\vdash_n \langle S \rangle\, \Gamma, t : A \vdash \Delta}{\vdash_n \langle S \rangle\, \Gamma, u : A' \vdash \Delta} \qquad\qquad \frac{\vdash_n \langle S \rangle\, \Gamma \vdash t : A, \Delta}{\vdash_n \langle S \rangle\, \Gamma \vdash u : A', \Delta}
$$

This proof transformation can be seen as yielding an explicit (admissible) proof of α-conversion in elements of the intended model (processes) *inside our logic* (to be distinguished from (α) in Lemma 4.2, which is about the syntax of the derivations). For this transformation to go through, explicit transpositions needed to be included both in the π-algebra and in the syntax of terms in formulas, to let

$$\frac{\langle S\rangle\, \Gamma \vdash \Delta}{\langle S\rangle\, \Gamma, u : \mathtt{T} \vdash \Delta}\;(\mathtt{T}\mathrm{L}) \qquad\qquad \frac{}{\langle S\rangle\, \Gamma \vdash u : \mathtt{T}, \Delta}\;(\mathtt{T}\mathrm{R})$$

$$\frac{\begin{array}{c}\langle S\rangle\, \Gamma, u : A \vdash \Delta\\ \langle S\rangle\, \Gamma, u : B \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : A \vee B \vdash \Delta}\;(\vee\mathrm{L}) \qquad\qquad \frac{\langle S\rangle\, \Gamma \vdash u : A, u : B, \Delta}{\langle S\rangle\, \Gamma \vdash u : A \vee B, \Delta}\;(\vee\mathrm{R})$$

$$\frac{\langle S\rangle\, \Gamma \vdash u : A, \Delta}{\langle S\rangle\, \Gamma, u : \neg A \vdash \Delta}\;(\neg\mathrm{L}) \qquad\qquad \frac{\langle S\rangle\, \Gamma, u : A \vdash \Delta}{\langle S\rangle\, \Gamma \vdash u : \neg A, \Delta}\;(\neg\mathrm{R})$$

$$\frac{\begin{array}{c}\langle S\rangle\, \Gamma, v : A \vdash \Delta\\ \langle S\rangle\, \Gamma, t : B \vdash \Delta \quad u \doteq_S v|t\end{array}}{\langle S\rangle\, \Gamma, u : A \| B \vdash \Delta}\;(\|\mathrm{L}) \qquad\qquad \frac{\begin{array}{c}[\mathcal{X} \textit{ and } \mathcal{Y} \textit{ not free in the conclusion}]\\ \langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma \vdash \mathcal{X} : A, \mathcal{Y} : B, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u : A \| B, \Delta}\;(\|\mathrm{R})$$

$$\frac{\begin{array}{c}[\mathcal{X} \textit{ not free in the conclusion}]\\ \langle S, u = (\boldsymbol{\nu} n)\mathcal{X}\rangle\, \Gamma \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : \copyright n \vdash \Delta}\;(\copyright\mathrm{L}) \qquad\qquad \frac{\langle S\rangle\, u \doteq_S (\boldsymbol{\nu} n)v}{\langle S\rangle\, \Gamma \vdash u : \copyright n, \Delta}\;(\copyright\mathrm{R})$$

$$\frac{\langle S\rangle\, \Gamma, v : A \vdash \Delta \quad u \to_S v}{\langle S\rangle\, \Gamma, u : \Box A \vdash \Delta}\;(\Box\mathrm{L}) \qquad\qquad \frac{\begin{array}{c}[X \textit{ not free in the conclusion}]\\ \langle S, u \to \mathcal{X}\rangle\, \Gamma \vdash \mathcal{X} : A, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u : \Box A, \Delta}\;(\Box\mathrm{R})$$

$$\frac{\langle S\rangle\, \Gamma, v : A \vdash \Delta}{\langle S\rangle\, \Gamma, u :!A \vdash \Delta}\;(!\mathrm{L}) \qquad\qquad \frac{\begin{array}{c}[\mathcal{X} \textit{ not free in the conclusion}]\\ \langle S\rangle\, \Gamma \vdash \mathcal{X} : A, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u :!A, \Delta}\;(!\mathrm{R})$$

$$\frac{\langle S\rangle\, \Gamma \vdash v : A, \Delta \quad \langle S\rangle\, \Gamma, v : B \vdash \Delta}{\langle S\rangle\, \Gamma, u : A \Mapsto B \vdash \Delta}\;(\Mapsto\mathrm{L}) \qquad\qquad \frac{\begin{array}{c}[\mathcal{X} \textit{ not free in the conclusion}]\\ \langle S\rangle\, \Gamma, \mathcal{X} : A \vdash \mathcal{X} : B, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u : A \Mapsto B, \Delta}\;(\Mapsto\mathrm{R})$$

$$\frac{\begin{array}{c}[x \textit{ not free in the conclusion}]\\ \langle S\rangle\, \Gamma, u : A \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : \exists x.A \vdash \Delta}\;(\exists\mathrm{L}) \qquad\qquad \frac{\langle S\rangle\, \Gamma \vdash u : A\{x \leftarrow n\}, \Delta}{\langle S\rangle\, \Gamma \vdash u : \exists x.A, \Delta}\;(\exists\mathrm{R})$$

$$\frac{\begin{array}{c}[X \textit{ not free in the conclusion}]\\ \langle S\rangle\, \Gamma, u : A \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : \exists X.A \vdash \Delta}\;(\exists^2\mathrm{L}) \qquad\qquad \frac{\langle S\rangle\, \Gamma \vdash u : A\{X \leftarrow B\}, \Delta}{\langle S\rangle\, \Gamma \vdash u : \exists X.A, \Delta}\;(\exists^2\mathrm{R})$$

$$\frac{\begin{array}{c}[\mathcal{X} \textit{ not free in the conclusion }]\\ n \,\#_S\, \mathsf{H}x.A \quad u \doteq_S (\boldsymbol{\nu} n)v\\ \langle S, u \doteq (\boldsymbol{\nu} n)\mathcal{X}\rangle\, \Gamma, \mathcal{X} : A\{x \leftarrow n\} \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma, u : \mathsf{H}x.A \vdash \Delta}\;(\mathsf{HL}) \qquad\qquad \frac{\begin{array}{c}n \,\#_S\, \mathsf{H}x.A \quad u \doteq_S (\boldsymbol{\nu} n)v\\ \langle S\rangle\, \Gamma \vdash v : A\{x \leftarrow n\}, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u : \mathsf{H}x.A, \Delta}\;(\mathsf{HR})$$

Figure 12: Inference Rules for derived connectives.

transpositions commute through derivations up to atomic instance of (Id) axioms (where cuts get finally removed). We can then prove

Theorem 4.5 (Cut Elimination) *If a sequent has a first-order derivation then it has a derivation without instances of the* (Cut) *rule.*

When developing the proof, we define an contraction-free variant of the basic proof system (*cf.* the system **G3c** in [15]), and identify a basic set of (size-preserving) inversion principles for our logic.

5 Examples

In this Section we go though a sequence of short examples to show how our logic is applicable to reasoning about distributed concurrent systems. We are necessarily brief here, and show only very elementary examples, but most interesting logical operators are covered.

5.1 A Simple Property

We show a simple derivation of the fact that $(A|B) \wedge \mathbf{0}$ entails A, meaning that if a process satisfies $(A|B) \wedge \mathbf{0}$ then it satisfies A. The intuition is that if a process P satisfies both $(A|B)$ and $\mathbf{0}$, then P is (structurally equivalent to) the $\mathbf{0}$ process, which is the same as $\mathbf{0}|\mathbf{0}$; hence $\mathbf{0}$ satisfies A (and B). We conclude that P satisfies A. This derivation illustrates: a property combining spatial and propositional operators; the use of constraint manipulation; and the use of one of the world rules, namely, $(S|\mathbf{0})$ corresponding to the "zero law" of π-calculus processes: if $P|Q \equiv \mathbf{0}$ then $P \equiv \mathbf{0}$.

5. $\langle S, u \doteq \mathcal{X}|\mathcal{Y}, u \doteq \mathbf{0}, \mathcal{X} \doteq \mathbf{0}\rangle\, \Gamma, \mathcal{X} : A, \mathcal{Y} : B \vdash u : A, \Delta$ (by (Id) since $u \doteq_S \mathcal{X}$)
4. $\langle S, u \doteq \mathcal{X}|\mathcal{Y}, u \doteq \mathbf{0}\rangle\, \Gamma, \mathcal{X} : A, \mathcal{Y} : B \vdash u : A, \Delta$ (by 5, (S|**0**) since $\mathcal{X}|\mathcal{Y} \doteq_S \mathbf{0}$)
3. $\langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : A, \mathcal{Y} : B, u : \mathbf{0} \vdash u : A, \Delta$ (by 4, (**0**L))
2. $\langle S\rangle\, \Gamma, u : (A|B), u : \mathbf{0} \vdash u : A, \Delta$ (by 3, (|L))
1. $\langle S\rangle\, \Gamma, u : (A|B) \wedge \mathbf{0} \vdash u : A, \Delta$ (by 2, ($\wedge$L))

Note that the proof is fairly simple, particularly if conducted bottom up. Most constraints are generated from the goal by using all the applicable left rules, and the final constraint $\mathcal{X} \doteq \mathbf{0}$ is generated by closing up the constraint set under deduction, via (S|**0**). Finally, (Id) involves a simple equivalence check in S. It is common for our derivations, when read bottom-up, to have this mechanical flavor.

5.2 Freshness

We show a derivation of the fact that $\neg \text{И} x.A$ entails $\text{И} x.\neg A$. This (and its converse) is a well-known property of $\text{И} x.A$ [7]; the purpose here is to show the use of the rules for freshness in a simple case. We abbreviate by $y \# \text{И} x.A$ the set of constraints $y \# \mathit{lfv}(\text{И} x.A)$.

6. $\langle S, y \# \text{И} x.A, u \doteq (\boldsymbol{\nu} y)\mathcal{X}\rangle\, \Gamma, u : A\{x \leftarrow y\} \vdash u : A\{x \leftarrow y\}, \Delta$ (by (Id) choose $y, \mathcal{X}$ fresh)
5. $\langle S, y \# \text{И} x.A, u \doteq (\boldsymbol{\nu} y)\mathcal{X}\rangle\, \Gamma \vdash u : A\{x \leftarrow y\}, u : \neg A\{x \leftarrow y\}, \Delta$ (by 6, ($\neg$ R))
4. $\langle S, y \# \text{И} x.A, u \doteq (\boldsymbol{\nu} y)\mathcal{X}\rangle\, \Gamma \vdash u : \text{И} x.A, u : \neg A\{x \leftarrow y\}, \Delta$ (by 5, (ИR))
3. $\langle S, y \# \text{И} x.A, u \doteq (\boldsymbol{\nu} y)\mathcal{X}\rangle\, \Gamma \vdash u : \text{И} x.A, u : \text{И} x.\neg A, \Delta$ (by 4, (ИR))
2. $\langle S, y \# \text{И} x.A, u \doteq (\boldsymbol{\nu} y)\mathcal{X}\rangle\, \Gamma, u : \neg \text{И} x.A \vdash u : \text{И} x.\neg A, \Delta$ (by 3, ($\neg$ L))
1. $\langle S\rangle\, \Gamma, u : \neg \text{И} x.A \vdash u : \text{И} x.\neg A, \Delta$ (by 2, (И) $y, \mathcal{X}$ not in conclusion)

We start with $A\{x \leftarrow y\}$ for a fresh y, instead of simply with A, so that we can apply (И) in the last step even when x occurs free in Γ, Δ. It is usually the case that an application of rules (И L) or (И R) is followed by an application of rule (И), to clean up the constraints. Note, however, that having (И) decoupled from (ИL) and (ИR) allow us to apply, in this case, (ИR) twice before applying (И).

Along similar lines, we can derive interesting properties combining $\text{И}x.A$ with spatial operators, for example the following one, which is important for deriving properties of the hiding quantifier (it takes about eight steps in each direction, but with a rather more involved set of constraints):

$$\langle S\rangle\, \Gamma, u : (\text{И}x.A)|(\text{И}x.B) \dashv\vdash u : \text{И}x.(A|B), \Delta$$

This derivation uses the world rule (S$\nu|$), which embeds a rather deep lemma about π-calculus structural congruence; namely, that if $(\boldsymbol{\nu} n)P \equiv Q|R$ then there exist P', Q' such that $P \equiv P|P''$ and $(\boldsymbol{\nu} n)P' \equiv Q$ and $(\boldsymbol{\nu} n)P'' \equiv R$.

5.3 Equivariance

In general, we have that an Aπ process P satisfies the formula $n \circledR A$ if $P \equiv (\boldsymbol{\nu} n)Q$, where Q is a process that satisfies A. Then n is a name which is hidden, and hence not free, in P. Therefore, the revelation operator has a useful meaning also in the special case $n \circledR \mathtt{T}$: the process P satisfies $n \circledR \mathtt{T}$ if and only if the name n is fresh in P (In Section 4.2 we introduced $\copyright n$ as an abbreviation for $\neg n \circledR A$). We can show than $A \wedge m \circledR \mathtt{T} \wedge n \circledR \mathtt{T}$ entails $\{m \leftrightarrow n\} \cdot A$:

$$\begin{array}{lr}
3.\ \langle \mathcal{Z} \doteq (\boldsymbol{\nu} n)\mathcal{X}, u \doteq (\boldsymbol{\nu} m)\mathcal{Y}\rangle\ \boldsymbol{(}m \leftrightarrow n\boldsymbol{)}\mathcal{Z} : \{m \leftrightarrow n\} \cdot A, \mathcal{X} : \mathtt{T}, \mathcal{Y} : \mathtt{T} \vdash \mathcal{Z} : \{m \leftrightarrow n\} \cdot A & \text{(Id)} \\
2.\ \langle \mathcal{Z} \doteq (\boldsymbol{\nu} n)\mathcal{X}, u \doteq (\boldsymbol{\nu} m)\mathcal{Y}\rangle\ \mathcal{Z} : A, \mathcal{X} : \mathtt{T}, \mathcal{Y} : \mathtt{T} \vdash \mathcal{Z} : \{m \leftrightarrow n\} \cdot A & \text{(by 3, } (\tau)\text{)} \\
1.\ \langle\rangle\ \mathcal{Z} : A \wedge m \circledR \mathtt{T} \wedge n \circledR \mathtt{T} \vdash u : \{m \leftrightarrow n\} \cdot A & \text{(by 2, } (\wedge\text{L and } \circledR\text{L}))
\end{array}$$

(Note the use of (Swap Erase) in step 3) This property can be interpreted as saying that, for any process P, if it satisfies A, it also satisfies $\{m \leftrightarrow n\} \cdot A$ for any fresh names m and n. This fact is a consequence of the equivariance property of the semantics: intuitively, if the name denoted by (say) m occurs in the formula A but not in the process P, then we would expect the name m to be irrelevant to the fact that P satisfies A. This means that if we swap in formula A the name m by any other fresh name n, we would expect that P would still satisfy it (since a fresh name is as good as any other fresh name). For example, the following provable sequent

$$\langle n \# p, m \# p\rangle\ n \circledR (p\langle n\rangle|\mathtt{T}) \wedge m \circledR \mathtt{T} \wedge n \circledR \mathtt{T} \vdash m \circledR (p\langle m\rangle|\mathtt{T})$$

says that if a process is about to send a fresh name on a public channel p, it can send any other fresh name as well.

5.4 Input

In our logic we have a primitive formula to observe messages, $n\langle m\rangle$, corresponding to the output operator of the asynchronous π-calculus. We do not have a corresponding input formula, but it can be expressed from output along the lines of [13]. The guarantee operator is crucial to this; recall that a process P satisfies $A \triangleright B$ if for any Q that satisfies A, we have that $P|Q$ satisfies B (this can be read out from ($\triangleright$R)). We say that P satisfies B "in presence" of any Q that satisfies A. We can take the following definition of input:

$$x(y).A \triangleq \forall y. x\langle y\rangle \triangleright \Diamond A$$

The intention is that a process satisfies the input specification $x(y).A$ if it performs an input over a given channel x of any name y (with y bound in A), and then satisfies the property A. The above definition says literally, that an input process is one that, in presence of any output message y over the given channel x, at the next step (after input) it behaves according to A.

It is then easy to verify that because of the adjunction between $|$ and $\triangleright$, input and output interact as expected in π-calculus communication, that is, $x\langle z\rangle|x(y).A$ entails $\Diamond A\{y\leftarrow z\}$:

4.2. $\langle S, u = \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : x\langle z\rangle \vdash \mathcal{X} : x\langle z\rangle, u : \Diamond A\{y\leftarrow z\}, \Delta$ (by (Id) choose $\mathcal{X}, \mathcal{Y}$ fresh)
4.1. $\langle S, u = \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : x\langle z\rangle, \mathcal{X}|\mathcal{Y} : \Diamond A\{y\leftarrow z\} \vdash u : \Diamond A\{y\leftarrow z\}, \Delta$ (by (Id) since $\mathcal{X}|\mathcal{Y} \doteq_S u$)
3. $\langle S, u = \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : x\langle z\rangle, \mathcal{Y} : x\langle z\rangle \triangleright \Diamond A\{y\leftarrow z\} \vdash u : \Diamond A\{y\leftarrow z\}, \Delta$ (by 4.1, 4.2, ($\triangleright$L))
2. $\langle S, u = \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : x\langle z\rangle, \mathcal{Y} : \forall y.x\langle y\rangle \triangleright \Diamond A \vdash u : \Diamond A\{y\leftarrow z\}, \Delta$ (by 3, ($\forall$L))
1. $\langle S\rangle\, \Gamma, u : x\langle z\rangle|(\forall y.x\langle y\rangle \triangleright \Diamond A) \vdash u : \Diamond A\{y\leftarrow z\}, \Delta$ (by 2, ($|$L) $\mathcal{X}, \mathcal{Y}$ not in conclusion)

5.5 Hiding

In Part I and Section 4.2 we defined a hiding quantifier: $\mathsf{H}x.A \triangleq \mathsf{И}x.x\circledR A$ which is related to π-calculus name restriction in an appropriate way; namely, that if process P satisfies formula $A\{x\leftarrow n\}$, then $(\boldsymbol{\nu} n)P$ satisfies $\mathsf{H}x.A$.

An interesting use of $\mathsf{H}x.A$ is in specifying "nonce generators", that is processes that output freshly generated names on a given channel. In π-calculus, a nonce generator can be written simply as $(\boldsymbol{\nu} n)nc\langle n\rangle$, for a given channel nc. A nonce generator over nc can then be specified by the following formula:

$$\mathcal{N}c \;\triangleq\; \mathsf{H}x.nc\langle x\rangle$$

We can show that, when a nonce generator interacts with an input, the result is the acquisition of a private name:

$$\langle S\rangle\, \Gamma, u : \mathcal{N}c|nc(y).A \vdash u : \Diamond \mathsf{H}z.A\{y\leftarrow z\}, \Delta$$

Before input we have a nonce generator $\mathcal{N}c$ separate from the input process. After one step, we have that the A part has acquired a name z; but noticeably this z is "hidden" within $A\{y\leftarrow z\}$ by the scope of the hiding quantifier. Hence the A part of the system has acquired, from the nonce generator, a private name not shared with other parts of the system (at least, not yet).

5.6 Inductive Definitions

We just sketch here our treatment of recursive formulas (alternatively they could be axiomatized directly, as in Part I). First, as shown in Section 4.2, we can combine the spatial operator $\triangleright$ with classical negation to obtain an operator $!A \triangleq (A \Rightarrow \mathtt{F}) \triangleright \mathtt{F}$ that has the meaning that A is valid (is satisfied by any process). Through second-order quantification, we can then define a least-fixpoint operator in a style similar to F-algebraic encodings. We can also define a dual greatest-fixpoint operator:

$$\mu X.A \;\triangleq\; \forall Y.((A\{X\leftarrow Y\} \Rrightarrow Y)) \Rightarrow Y \;\; \textit{(where } Y \textit{ is not free in } A\textit{)}$$
$$\nu X.A \;\triangleq\; \neg\mu X.\neg A\{X\leftarrow\neg X\}$$

these definitions turn out to enjoy the expected properties of recursive formulas, in the form of derivable left and right rules; for example the derivable rule (νR) corresponds to a coinduction principle:

$$\frac{\begin{array}{c}[\mathcal{X} \textit{ is not free in the conclusion}]\\ \langle S\rangle\, \Gamma, \mathcal{X} : B \vdash \mathcal{X} : A\{X\leftarrow B\}, \Delta \quad \langle S\rangle\, \Gamma \vdash u : B, \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash u : \nu X.A, \Delta}\;(\nu\mathrm{R})$$

The folding and unfolding of $\mu X.A$ and $\nu X.A$ can be derived under an assumption of monotonicity of $A\{X\}$; this can be expressed via the ! operator. As an example of the use of recursion, we can specify a recursive nonce generator (a process producing an unbounded number of fresh names) as follows:

$$\mathcal{UN}c \;\triangleq\; \nu X.\mathcal{N}c|X$$

By a standard coinductive argument, we can then show that $\mathcal{UN}c|\mathcal{UN}c$ entails $\mathcal{UN}c$. This is simple but significant: it means that (without any knowledge of the π-calculus implementation) two recursive nonce generators running in parallel behave like a single recursive nonce generator; in particular, the two generators do not risk generating independently the same name twice.

6 Conclusion

We have presented a sequent calculus that has a direct interpretation in terms of distributed concurrent behaviors, including notions of resource hiding. We believe we have obtained a unique combination of, on one hand, good proof-theoretical structures and properties, and, on the other hand, direct applicability to concurrency. These twin aims have driven us towards a "many worlds" formulation of modal sequents that has been able to accommodate a wide range of unusual but strongly motivated logical constructions.

References

[1] L. Caires. *A Model for Declarative Programming and Specification with Concurrency and Mobility*. PhD thesis, Dept. de Informática, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, 1999.

[2] L. Caires and L. Cardelli. A Spatial Logic for Concurrency (Part I). In B.C. Pierce N. Kobayashi, editor, *Proceedings of the 10th Symposium on Theoretical Aspects of Computer Science (TACS 2001)*, volume 2215 of *Lecture Notes in Computer Science*, pages 1–30. Springer-Verlag, 2001.

[3] L. Caires and L. Monteiro. Verifiable and Executable Specifications of Concurrent Objects in $\mathcal{L}_\pi$. In C. Hankin, editor, *Programming Languages and Systems: Proceedings of the 7th European Symp. on Programming (ESOP 1998)*, number 1381 in Lecture Notes in Computer Science, pages 42–56. Springer-Verlag, 1998.

[4] L. Cardelli and A. D. Gordon. Anytime, Anywhere. Modal Logics for Mobile Ambients. In *27th ACM Symposium on Principles of Programming Languages*, pages 365–377. ACM, 2000.

[5] L. Cardelli and A. D. Gordon. Logical Properties of Name Restriction. In S. Abramsky, editor, *Typed Lambda Calculi and Applications*, number 2044 in Lecture Notes in Computer Science. Springer-Verlag, 2001.

[6] M. Dam. Relevance Logic and Concurrent Composition. In *Proceedings, Third Annual Symposium on Logic in Computer Science*, pages 178–185, Edinburgh, Scotland, 5–8 July 1988. IEEE Computer Society.

[7] M. J. Gabbay and A. M. Pitts. A New Approach to Abstract Syntax Involving Binders. In *14th Annual Symposium on Logic in Computer Science*, pages 214–224. IEEE Computer Society Press, Washington, 1999.

[8] M. Hennessy and R. Milner. Algebraic laws for Nondeterminism and Concurrency. *JACM*, 32(1):137–161, 1985.

[9] D. Kozen. Results on the Propositional μ-Calculus. *TCS*, 27(3):333–354, 1983.

[10] P. O'Hearn and D. Pym. The Logic of Bunched Implications. *The Bulletin of Symbolic Logic*, 5(2):215–243, 1999.

[11] A. Pitts. Nominal Logic: A First Order Theory of Names and Binding. In B.C. Pierce N. Kobayashi, editor, *Proceedings of the 10th Symposium on Theoretical Aspects of Computer Science (TACS 2001)*, volume 2215 of *Lecture Notes in Computer Science*, pages 219–235. Springer-Verlag, 2001.

[12] G. Plotkin and M. Abadi. A logic for parametric polymorphism. In M. Bezem and J. F. Groote, editors, *International Conference on Typed Lambda Calculi and Applications*, number 664 in Lecture Notes in Computer Science, pages 361–375, Utrecht, The Netherlands, March 1993. Springer-Verlag. TLCA'93.

[13] D. Sangiorgi. Extensionality and Intensionality of the Ambient Logics. In *28th Annual Symposium on Principles of Programming Languages*, pages 4–13. ACM, 2001.

[14] A. Simpson. *The Proof Theory and Semantics of Intuitionistic Modal Logic*. Ph.D. thesis, Dept. of Computer Science, Edingburgh University, 1994.

[15] A. Troelstra and H. Schwichtenberg. *Basic Proof Theory*. Cambridge University Press, 2000.

[16] Luca Viganò. *Labelled Non-Classical Logics*. Kluwer Academic Publishers, Dordrecht, 2000.

7 Appendix

7.1 Soundness (Sequent Calculus)

An inference rule is *sound* if the sequent in the conclusion is valid provided all the sequents and assertions occurring as premises are valid (see Definitions 2.10 and 3.5). Note that familiarity with the semantics given in [2] is required before reading this section.

Theorem 3.6 [Soundness] *All derivable sequents are valid in* $\mathrm{A}\pi$.

Proof. We show that all inference rules are sound. Cases of (FL), (FR), ($\wedge$L), ($\wedge$R), ($\Rightarrow$L) and ($\Rightarrow$R) are standard.

- (Id) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma, u : A \vdash t : A', \Delta$ such that I satisfies S and all of $\Gamma, u : A$. Then $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(A)]\!]_v$. Since $A \equiv_S A'$ and $u \doteq_S t$ then $I_{\mathcal{L}}(A) = I_{\mathcal{L}}(A')$ and $I(u) \equiv I(t)$. Hence, by the main theorem of Part I, we conclude $I_{\mathcal{P}}(t) \in [\![I_{\mathcal{L}}(A')]\!]_v$.
- (Cut) Similar to (Id).
- (S$\boldsymbol{\nu}|$) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma \vdash \Delta$ in the conclusion such that I satisfies S and $\mathcal{J}$ satisfies all of Γ. In particular, we have $(\boldsymbol{\nu} I(x))I_{\mathcal{P}}(u) \equiv I_{\mathcal{P}}(t)|I_{\mathcal{P}}(v)$. By Proposition (Part I)2.13(2), there are processes P and Q such that $t \equiv (\boldsymbol{\nu} I(x))P$, $v \equiv (\boldsymbol{\nu} I(x))Q$, and $I_{\mathcal{P}}(u) \equiv P|Q$. Let $\mathcal{J}' \triangleq (v, I')$ where $I' = I\{\mathcal{X}\leftarrow P\}\{\mathcal{Y}\leftarrow Q\}$. I' satisfies $\langle S, u \doteq \mathcal{X}|\mathcal{Y}, (\boldsymbol{\nu} x)\mathcal{X} \doteq t, (\boldsymbol{\nu} x)\mathcal{Y} \doteq v\rangle$. Since $\mathcal{X}$ and $\mathcal{Y}$ do not occur in Γ and Δ, we have that $\mathcal{J}'$ satisfies all of Γ, hence by validity of the premises it also satisfies some of Δ. So also $\mathcal{J}$ satisfies some of Δ.
- (S$-$). Like with (S$\boldsymbol{\nu}|$) above, soundness is a consequence of the inversion properties of Proposition (Part I)2.13(2).
- ($\mathbf{0}$R) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma \vdash \Delta$ in the conclusion, and assume that I satisfies S. Hence $I_{\mathcal{P}}(u) \equiv \mathbf{0}$, thus $I_{\mathcal{P}}(u) \in [\![\mathbf{0}]\!]_v$.
- ($\mathbf{0}$L) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma \vdash \Delta$ in the conclusion, and assume that $\mathcal{J}$ satisfies all of $\Gamma, u : \mathbf{0}$. Hence $I_{\mathcal{P}}(u) \equiv \mathbf{0}$, and I satisfies $\langle S, u \doteq \mathbf{0}\rangle$. So, by validity of the premise, $\mathcal{J}$ satisfies some of Δ.
- ($|$R) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma \vdash \Delta$ in the conclusion, and assume that I satisfies S and $\mathcal{J}$ satisfies all of Γ. By assumption, $I_{\mathcal{P}}(u) \equiv I_{\mathcal{P}}(v)|I_{\mathcal{P}}(t)$. If $\mathcal{J}$ satisfies some of Δ, we have the conclusion. Otherwise, by validity of the premises, we must have $I_{\mathcal{P}}(v) \in [\![I_{\mathcal{L}}(A)]\!]_v$ and $I_{\mathcal{P}}(t) \in [\![I_{\mathcal{L}}(B)]\!]_v$. From that, we conclude $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(A|B)]\!]_v$.
- ($|$L) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent in the conclusion, and assume that I satisfies S and $\mathcal{J}$ satisfies all of $\Gamma, u : A|B$. Thus, there are P and Q such that $I_{\mathcal{P}}(u) \equiv P|Q$, $P \in [\![I_{\mathcal{L}}(A)]\!]_v$ and $Q \in [\![I_{\mathcal{L}}(B)]\!]_v$. Let $\mathcal{J}' \triangleq (v, I')$ where $I' \triangleq I\{\mathcal{X}\leftarrow P\}\{\mathcal{Y}\leftarrow Q\}$: then I' satisfies $\langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle$ and $\mathcal{J}'$ satisfies all of $\Gamma, \mathcal{X} : A, \mathcal{Y} : B$. To conclude, note that by assumption $\mathcal{J}'$ satisfies some of Δ, and that $\mathcal{J}'$ agrees with $\mathcal{J}$ on Δ.
- ($\triangleright$R) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent in the conclusion, and assume that I satisfies S and $\mathcal{J}$ satisfies all of Γ. Pick any process $P \in [\![I_{\mathcal{L}}(A)]\!]_v$. Since $\mathcal{X}$ does not occur in the conclusion, the interpretation $\mathcal{J}^P \triangleq (v, I^P)$ where $I^P \triangleq I\{\mathcal{X}\leftarrow P\}$) also satisfies S and all of $\Gamma, \mathcal{X} : A$. By assumption, $I^P(v) \in [\![{I^P}_{\mathcal{L}}(B)]\!]_v = [\![I_{\mathcal{L}}(B)]\!]_v$. But $I^P(v) \equiv P|I^P(u) = P|I_{\mathcal{P}}(u)$. Hence $P|I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(B)]\!]_v$, for all processes $P \in [\![I_{\mathcal{L}}(A)]\!]_v$. We conclude $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(A \triangleright B)]\!]_v$.
- ($\triangleright$L) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent in the conclusion, and assume that I satisfies S and $\mathcal{J}$ satisfies all of $\Gamma, u : A \triangleright B$. Thus, for all processes P such that $P \in [\![I_{\mathcal{L}}(A)]\!]_v$

we have that $P|I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(B)]\!]_v$. Since $\mathcal{J}$ satisfies all of Γ, by validity of the left premise either $\mathcal{J}$ satisfies $t : A$ or $\mathcal{J}$ satisfies some of Δ. In the latter case, we are done. Otherwise, $I_{\mathcal{P}}(t) \in [\![I_{\mathcal{L}}(A)]\!]_v$. Then $I_{\mathcal{P}}(t|u) \in [\![I_{\mathcal{L}}(B)]\!]_v$, hence $\mathcal{J}$ satisfies all of $\Gamma, t|u : B$. By validity of the right premise, we also conclude that $\mathcal{J}$ satisfies some of Δ.

- ($\Diamond$R) and ($\Diamond$L) By Lemma 2.11(2).
- (®L), (®R), ($\oslash$L), and ($\oslash$R) Proofs similar to (|R), (|L), ($\triangleright$L) and ($\triangleright$R) above.
- ($\forall$R) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma \vdash u : \forall x.A, \Delta$ in the conclusion such that I satisfies S and $\mathcal{J}$ satisfies all of Γ. Pick any name n and define $I^n \triangleq I\{x\leftarrow n\}$; $\mathcal{J}^n = (v, I^n)$ is then an interpretation for the sequent $\langle S\rangle\, \Gamma \vdash u : A, \Delta$ in the premise. Note that for all names n, I^n satisfies S and $\mathcal{J}^n$ satisfies all of Γ, since x does not occur free in the conclusion of the rule. Hence, by validity of the premise, $\mathcal{J}^n$ satisfies some of $u : A, \Delta$, for all n. Now, suppose there is an interpretation $\mathcal{J}^n$ that satisfies some of Δ. Then also $\mathcal{J}$ satisfies some of Δ since x is not free in Δ, and we have the conclusion. Otherwise, we must have $I^n(u) \in [\![I^n_{\mathcal{L}}(A)]\!]_v$ for all names n. But then, $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(A\{x\leftarrow n\})]\!]_v$ for all n, that is, $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(\forall x.A)]\!]_v$.
- ($\forall$L) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma, u : \forall x.A \vdash \Delta$ in the conclusion such that I satisfies S, $\mathcal{J}$ satisfies all of $\Gamma, u : \forall x.A$. Hence, we have $I_{\mathcal{P}}(u)[\![I_{\mathcal{L}}\{x\leftarrow x\}(A)\{x\leftarrow p\}]\!]_v$ for all names p. Since $y \in fv(N)$ we have $I_{\mathcal{L}}(y) = m$ for some name m. Hence, $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}\{x\leftarrow x\}(A)\{x\leftarrow m\}]\!]_v = [\![I_{\mathcal{L}}\{x\leftarrow m\}(A)]\!]_v$, that is, $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(A\{x\leftarrow y\})]\!]_v$. By validity of the sequent in the premise, we conclude that $\mathcal{J}$ satisfies some of Δ.
- ($\forall^2$R) and ($\forall^2$L) The semantics of $\forall X.A$ is given by $[\![\forall X.A]\!]_v \triangleq \bigcap_{\Phi\in\mathbb{P}_-}[\![A]\!]_{v[X\leftarrow\Phi]}$. The proof is then similar to ($\forall$L) and ($\forall$R) above, making use of Lemma 4.23 of Part I.
- (И) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma \vdash \Delta$ such that I satisfies S and $\mathcal{J}$ satisfies all of Γ. Now, let $P = I_{\mathcal{P}}(u)$ and pick any name $n \notin fn(P)$ such that $n \neq I(y)$ for all $y \in fv(N)$ and $n \notin supp(v(X))$, for all $X \in fpv(N)$. When then have $I_{\mathcal{P}}(u) \equiv (\boldsymbol{\nu} n)I_{\mathcal{P}}(u)$. Define $I' \triangleq I\{x\leftarrow n\}\{X\leftarrow P\}$. Hence $\mathcal{J}' = (v, I')$ is an interpretation for the sequent in the premise, where I' satisfies $\langle S, u \doteq (\boldsymbol{\nu} x)X, x \# N\rangle$ and $\mathcal{J}'$ satisfies all of Γ (since x and X are fresh). By validity of such sequent, we conclude that $\mathcal{J}'$ satisfies some of Δ. Since $\mathcal{J}'$ agrees with $\mathcal{J}$ on Δ, we conclude that $\mathcal{J}$ satisfies some of Δ.
- (τ) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma, u : A \vdash \Delta$ such that I satisfies S and $\mathcal{J}$ satisfies all of $\Gamma, u : A$. Hence $I(u) \in [\![I_{\mathcal{L}}(A)]\!]_v$. By validity of the premises, we have that $I_{\mathcal{L}}(m), I_{\mathcal{L}}(n) \notin supp(v(X))$, for all $X \in fpv(A)$.

  By the main theorem of Part I, $\{I_{\mathcal{L}}(m)\leftrightarrow I_{\mathcal{L}}(n)\}I(u) \in [\![\{I_{\mathcal{L}}(m)\leftrightarrow I_{\mathcal{L}}(n)\}I_{\mathcal{L}}(A)]\!]_v$. By Lemma 7.3(1), this implies $I((m\leftrightarrow n)u) \in [\![I_{\mathcal{L}}(\{m\leftrightarrow n\}\cdot A)]\!]_v$. Hence, by assumption, $\mathcal{J}$ satisfies some of Δ.
- (ИR) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma \vdash u : \text{И}x.A, \Delta$ such that I satisfies S and $\mathcal{J}$ satisfies all of Γ. By validity of the premise, $\mathcal{J}$ satisfies some of $u : A\{x\leftarrow n\}, \Delta$. If $\mathcal{J}$ satisfies some of Δ the proof is concluded. Otherwise, $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(A\{x\leftarrow n\})]\!]_v$. By assumption, the assertion $u \doteq_S (\boldsymbol{\nu} n)v$ is valid, hence $I_{\mathcal{P}}(u) \equiv (\boldsymbol{\nu} I_{\mathcal{L}}(n))I_{\mathcal{P}}(v)$. So, $I_{\mathcal{L}}(n) \notin fn(I_{\mathcal{P}}(u))$. Moreover, since $n \#_N \text{И}x.A$, by Lemma 7.5 we have $I_{\mathcal{L}}(n) \notin fn^v(I_{\mathcal{L}}(\text{И}x.A))$. Since $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(A\{x\leftarrow I_{\mathcal{L}}(n)\})]\!]_v$, we conclude $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(\text{И}x.A)]\!]_v$.
- (ИL) Let $\mathcal{J} = (v, I)$ be an interpretation for the sequent $\langle S\rangle\, \Gamma, u : \text{И}x.A \vdash \Delta$ such that I satisfies S and $\mathcal{J}$ satisfies all of $\Gamma, u : \text{И}x.A$. By the Gabbay-Pitts property (Proposition (Part I)6.1), for all names $p \in \Lambda$ such that $p \notin fn(I_{\mathcal{P}}(u)) \cup fn^v(I_{\mathcal{L}}(\text{И}x.A))$ we have $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(A\{x\leftarrow p\})]\!]_v$. Like in the case above for (ИR), we can verify that $I_{\mathcal{L}}(n) \notin fn(I_{\mathcal{P}}(u)) \cup fn^v(I_{\mathcal{L}}(\text{И}x.A))$. Hence, we

have $I_{\mathcal{P}}(u) \in [\![I_{\mathcal{L}}(A\{x\leftarrow I_{\mathcal{L}}(n)\})]\!]_v = [\![I_{\mathcal{L}}(A\{x\leftarrow n\})]\!]_v$. Since the premise of the rule is valid by assumption, we conclude that $\mathcal{J}$ satisfies some of Δ. ■

7.2 Some Operations on Formulas

We collect here the definitions of some elementary operations on formulas and related results.

Definition 7.1 (Free terms) *If A is a formula, we define the set $ft(A)$ of* free terms *in A by*

$$\begin{array}{llll}
ft(A \wedge B) = ft(A|B) = & & ft(\mathbf{0}) = ft(\mathtt{F}) & \triangleq \emptyset \\
ft(A \triangleright B) = ft(A \Rightarrow B) & \triangleq ft(A) \cup ft(B) & ft(m\langle n\rangle) & \triangleq \{m, n\} \\
ft(\forall x.A) = ft(\text{И}x.A) & \triangleq ft(A) \setminus \{x\} & ft(\forall X.A) & \triangleq ft(A) \setminus \{X\} \\
ft(m\circledR A) = ft(A\oslash m) & \triangleq \{m\} \cup ft(A) & ft(X) & \triangleq \{X\}
\end{array}$$

If $A[n_1, \ldots, n_k]$ is a formula with free terms $n_1, \ldots, n_k$, then $\tau{\cdot}A[n_1, \ldots, n_k]$ is the formula $A[\tau{\cdot}n_1, \ldots, \tau{\cdot}n_k]$ (taking care, by an α-conversion step, that no name variable in τ becomes bound by the context $A[-, \ldots, -]$):

Definition 7.2 (Transposition of Formulas) *Given a formula A and transposition $\tau = \{m{\leftrightarrow}n\}$ we denote by $\tau{\cdot}A$ the formula $\tau{\cdot}A_\emptyset$, where for any set of variables V, formula A and name term m, the term $\tau{\cdot}m_V$ and the formula $\tau{\cdot}A_V$ are defined by*

$$\begin{array}{lclllcl}
\tau{\cdot}x_V & \triangleq & x \quad (\textit{if } x \in V) & & \tau{\cdot}p_V & \triangleq & (m \leftrightarrow n)p \quad (\textit{if } p \notin V) \\
\\
\tau{\cdot}(A \wedge B)_V & \triangleq & \tau{\cdot}A_V \wedge \tau{\cdot}B_V & & \tau{\cdot}\mathtt{F}_V & \triangleq & \mathtt{F} \\
\tau{\cdot}(A \Rightarrow B)_V & \triangleq & \tau{\cdot}A_V \Rightarrow \tau{\cdot}B_V & & \tau{\cdot}\mathbf{0}_V & \triangleq & \mathbf{0} \\
\tau{\cdot}(A|B)_V & \triangleq & \tau{\cdot}A_V|\tau{\cdot}B_V & & \tau{\cdot}\Diamond A_V & \triangleq & \Diamond\tau{\cdot}A_V \\
\tau{\cdot}(A \triangleright B)_V & \triangleq & \tau{\cdot}A_V \triangleright \tau B_V & & \tau{\cdot}(\text{И}x.A) & \triangleq & \text{И}x.(\tau{\cdot}A)_{V,x} \\
\tau{\cdot}p\langle q\rangle_V & \triangleq & \tau{\cdot}p_V\langle\tau{\cdot}q_V\rangle & & \tau{\cdot}(\forall x.A)_V & \triangleq & \forall x.(\tau{\cdot}A)_{V,x} \\
\tau{\cdot}(p\circledR A)_V & \triangleq & \tau{\cdot}p_V\circledR\tau{\cdot}A & & \tau{\cdot}X_V & \triangleq & X \\
\tau{\cdot}(A\oslash p)_V & \triangleq & \tau{\cdot}A_V\oslash\tau{\cdot}p_V & & \tau{\cdot}(\forall X.A)_V & \triangleq & \forall X.(\tau{\cdot}A)_V
\end{array}$$

The auxiliary set V records bound occurrences, so that transpositions on them get ignored. We write $\tau{\cdot}A$ when τ is an arbitrary transposition, and ρ to represent an arbitrary sequence of transpositions. If A is a simple formula, we note by $A\{m{\leftrightarrow}n\}$ the interchange of the pure names m and n ($m, n \in \Lambda$) in A. For the relations $\equiv_S$ (introduced in Definition 3.1), and $\#_S$ (introduced in Definition 3.2) the following properties can be shown.

Lemma 7.3 *Let $\mathcal{J} = (v, I)$ be a logical interpretation and S a constraint theory. For all formulas A, B and name terms m, n we have:*

1. $I_{\mathcal{L}}(\{m{\leftrightarrow}n\}{\cdot}A) = I_{\mathcal{L}}(A)\{I_{\mathcal{L}}(m){\leftrightarrow}I_{\mathcal{L}}(n)\}$.
2. *If $\mathcal{J}$ satisfies S and $A \equiv_S B$ then $I_{\mathcal{L}}(A) = I_{\mathcal{L}}(B)$.*
3. *For every π-substitution σ, if $A \equiv_S B$ then $\sigma(A) \equiv_{\sigma(S)} \sigma(B)$.*
4. *For every π-substitution σ, we have $\sigma(\{m{\leftrightarrow}n\}{\cdot}A) = \{\sigma(m){\leftrightarrow}\sigma(n)\}{\cdot}\sigma(A)$.*

Lemma 7.4 *For all formulas A and name terms m, n we have*

1. *If $n \#_S A$ and σ is a π-substitution then $\sigma(n) \#_{\sigma(S)} \sigma(A)$.*

2. *For all name terms p, $p \#_S A$ implies $(m{\leftrightarrow}n)p \#_S \{m{\leftrightarrow}n\}{\cdot}A$.*

Lemma 7.5 *Let $\mathcal{J} = (v, I)$ be an interpretation and S a constraint theory. For all formulas A such that $fpv(A) \subseteq \mathfrak{D}(v)$, and name terms n, if $\mathcal{J}$ satisfies S and $n \#_S A$ then $I_{\mathcal{L}}(n) \notin fn^v(I_{\mathcal{L}}(A))$.*

Lemma 7.3(1) motivates in part the need of explicitly transposing names in formulas. In fact, such property would fail if $\{x{\leftrightarrow}y\}{\cdot}A$ is understood as the meta-level (capture avoiding) transposition. Concerning Definition 3.2, note that we may well have *e.g.*, $x \in fv(A)$ and $x \#_S A$. In any case, Lemma 7.5 fully justifies the definition.

7.3 Admissible Proof Principles

Lemma 4.2(Basic) *The following size-preserving proof principles are admissible:*

$$\frac{[\ \varphi' \text{ is not free in the premise }]\quad \vdash_n \langle S\rangle\, \Gamma \vdash \Delta}{\vdash_n \langle S\{\varphi{\leftarrow}\varphi'\}\rangle\, \Gamma\{\varphi{\leftarrow}\varphi'\} \vdash \Delta\{\varphi{\leftarrow}\varphi'\}}\ \text{(Ren)} \qquad \frac{[\ \Gamma \equiv_\alpha \Gamma' \text{ and } \Delta \equiv_\alpha \Delta'\ ]\quad \vdash_n \langle S\rangle\, \Gamma \vdash \Delta}{\vdash_n \langle S\rangle\, \Gamma' \vdash \Delta'}\ (\alpha)$$

$$\frac{\vdash_n \langle S\rangle\, \Gamma \vdash \Delta}{\vdash_n \langle S, S'\rangle\, \Gamma, \Gamma' \vdash \Delta, \Delta'}\ \text{(W)} \qquad \frac{\vdash_n \langle S, c\rangle\, \Gamma \vdash \Delta \quad S \vdash c}{\vdash_n \langle S\rangle\, \Gamma \vdash \Delta}\ \text{(CS)}$$

$$\frac{\vdash_n \langle S\rangle\, \Gamma \vdash \Delta}{\vdash_n \langle S\{\mathcal{X}{\leftarrow}u\}\rangle\, \Gamma\{\mathcal{X}{\leftarrow}u\} \vdash \Delta\{\mathcal{X}{\leftarrow}u\}}\ (\text{Inst}\mathcal{I}) \qquad \frac{\vdash_n \langle S\rangle\, \Gamma \vdash \Delta}{\vdash_n \langle S\{x{\leftarrow}m\}\rangle\, \Gamma\{x{\leftarrow}m\} \vdash \Delta\{x{\leftarrow}m\}}\ (\text{Inst}\mathcal{N})$$

Proof.

(CS) By induction on the structure of derivations, using Lemma 2.8(2).

(Ren) and (α) By simultaneous induction on the structure of of derivations.

(W) By induction on the structure of derivations, using Lemma 2.8(1) to show that provability of constraint premises is preserved.

(Inst$\mathcal{I}$) For clarity, we abbreviate the substitution $\{x{\leftarrow}u\}$ by σ. Proof by induction on the structure of derivations, using Lemma 2.8(3) to show that $u \doteq_S v$ implies $\sigma(u) \doteq_{\sigma(S)} \sigma(v)$ in all rule instances with assertions $u \doteq_S v$ as premises, and likewise for premises of the form $u \rightarrow_S v$ or $x \#_S A$. The most interesting cases are the ones which introduce process eigenvariables, *e.g.*,

- **(Case of (|L))** $\langle S\rangle\, \Gamma, u : A|B \vdash \Delta$ is concluded from $\langle S, u \doteq \mathcal{X}'|\mathcal{Y}'\rangle\, \Gamma, \mathcal{X}' : A, \mathcal{Y}' : B \vdash \Delta$. By (Ren), there is a derivation of $\langle S, u \doteq \mathcal{X}''|\mathcal{Y}''\rangle\, \Gamma, \mathcal{X}'' : A, \mathcal{Y}'' : B \vdash \Delta$, where $\mathcal{X}''$ and $\mathcal{Y}''$ are distinct from $\mathcal{X}$ and do not belong to $afv(u)$. By induction hypothesis, we have $\langle\sigma(S), u \doteq \mathcal{X}''|\mathcal{Y}''\rangle\, \Gamma, \mathcal{X}'' : \sigma(A), \mathcal{Y}'' : \sigma(B) \vdash \sigma(\Delta)$. We then conclude by (|L).

(Inst$\mathcal{N}$) For clarity, we abbreviate the substitution $\{x{\leftarrow}m\}$ by σ. The proof proceeds by induction on the structure of derivations and case analysis on the last rule used, using Lemma 2.8(3) to show that all constraints that occur as premises of rule instances in the derivation are preserved. We present a detailed proof for one of the spatial rules, the (И) and (τ) rules, and all the quantifier rules. In each case, note that the structure of the derivation is indeed preserved by the transformation.

- **(Case of (|R))** $\langle S\rangle\, \Gamma \vdash u : A|B, \Delta$ is concluded from $\langle S\rangle\, \Gamma \vdash t : A, \Delta$ and $\langle S\rangle\, \Gamma \vdash v : B, \Delta$ and $u \doteq_S t|v$. By induction hypothesis, we have $\langle\sigma(S)\rangle\, \Gamma \vdash \sigma(t) : \sigma(A), \sigma(\Delta)$ and $\langle\sigma(S)\rangle\, \Gamma \vdash \sigma(v) : \sigma(B), \sigma(\Delta)$. By Lemma 2.8, we have $\sigma(u) \doteq_{\sigma(S)} \sigma(t)|\sigma(v)$. We conclude by (|R).
- **(Case of (τ))** Suppose the instance of (τ) is not simple. Then $\langle S\rangle\, \Gamma, u : A \vdash \Delta$ is obtained by (τ) from $\langle S\rangle\, \Gamma, (p{\leftrightarrow}n)u : \{p{\leftrightarrow}n\}{\cdot}A \vdash \Delta$ where $p, n \#_S fpv(A)$.

By Lemma 2.8(3), $\sigma(p), \sigma(n) \#_{\sigma(S)} \mathit{fpv}(A)$. By induction hypothesis, there is a derivation of $\langle\sigma(S)\rangle\, \Gamma, \sigma((p \leftrightarrow n)u) : \sigma(\{p \leftrightarrow n\} \cdot A) \vdash \sigma(\Delta)$. We conclude by (α) and (τ), since $\sigma(\{p \leftrightarrow n\} \cdot A) \equiv_\alpha \{\sigma(p) \leftrightarrow \sigma(n)\} \cdot \sigma(A)$ and $\sigma((p \leftrightarrow n)u) = (\sigma(p) \leftrightarrow \sigma(n))\sigma(u)$. Otherwise, the instance of (τ) is simple. Then, we can easily obtain a derivation of unit size for $\langle\sigma(S)\rangle\, \sigma(\Gamma), \sigma(u) : \sigma(A) \vdash \sigma(\Delta)$ by instantiating every sequent in the given derivation with σ.

- **(Case of (И))** $\langle S\rangle\, \Gamma \vdash \Delta$ is obtained by (И) from $\langle S, u \doteq (\boldsymbol{\nu} z)X, z \# N\rangle\, \Gamma \vdash \Delta$, where z and X do not occur free in the conclusion and u, and N is a finite set of names not containing z. By (α) we may assume that $z \neq x$ and $z \notin \mathit{afv}(m)$.

  By induction hypothesis, we have $\langle\sigma(S), \sigma(u) \doteq (\boldsymbol{\nu} z)X, z \# \sigma(N)\rangle\, \sigma(\Gamma) \vdash \sigma(\Delta)$.

  Let $M = \mathit{afv}(\sigma(N))$. By (W), we have $\langle\sigma(S), \sigma(u) \doteq (\boldsymbol{\nu} z)X, z \# \sigma(N), z \# M\rangle\, \sigma(\Gamma) \vdash \sigma(\Delta)$. Write $S' \triangleq \langle\sigma(S), \sigma(u) \doteq (\boldsymbol{\nu} z)X, z \# M\rangle$. Since $M = \mathit{afv}(\sigma(N))$ and $z \#_{S'} M$, we can verify that $z \#_{S'} \sigma(N)$. By (CS) we have $\langle\sigma(S), \sigma(u) \doteq (\boldsymbol{\nu} z)X, z \# M\rangle\, \sigma(\Gamma) \vdash \sigma(\Delta)$.

  Now, note that z does not occur free in $\sigma(S)$, in $\sigma(\Gamma)$, u or M, or in $\sigma(\Delta)$, because it does not occur free in m, nor in the conclusion of the original sequent. Hence, by (И), we obtain the conclusion $\langle\sigma(S)\rangle\, \sigma(\Gamma) \vdash \sigma(\Delta)$.

- **(Case of (∀R))** The sequent $\langle S\rangle\, \Gamma \vdash u : \forall z.A, \Delta$ is concluded from the sequent $\langle S\rangle\, \Gamma \vdash u : A\{z \leftarrow y\}, \Delta$, where y does not occur free in the conclusion. By induction hypothesis, we have $\langle\sigma(S)\rangle\, \sigma(\Gamma) \vdash \sigma(v) : \sigma(A\{z \leftarrow y\}), \sigma(\Delta)$. We have $\sigma\{z \leftarrow z\}(A)\{z \leftarrow y\} \equiv_\alpha \sigma(A\{z \leftarrow y\})$. Thus, by (α) we have $\langle\sigma(S)\rangle\, \sigma(\Gamma) \vdash \sigma(v) : \sigma\{z \leftarrow z\}(A)\{z \leftarrow y\}, \sigma(\Delta)$. By (∀R) we finally conclude $\langle\sigma(S)\rangle\, \sigma(\Gamma) \vdash \sigma(v) : \sigma(\forall z.A), \sigma(\Delta)$, since $\sigma(\forall z.A) = \forall z.\sigma\{z \leftarrow z\}(A)$.

- **(Case of (ИR))** We have the sequent $\langle S\rangle\, \Gamma \vdash u : \text{И} z.A, \Delta$, concluded by (И) from a derivation of $\langle S\rangle\, \Gamma \vdash u : A\{z \leftarrow n\}, \Delta$, where $u \doteq_S (\boldsymbol{\nu} n)v$ and $n \#_S \text{И} z.A$. By induction hypothesis, we have $\langle\sigma(S)\rangle\, \sigma(\Gamma) \vdash \sigma(u) : \sigma(A\{z \leftarrow n\}), \sigma(\Delta)$.

  By Lemma 2.8, we have $\sigma(u) \doteq_{\sigma(S)} (\boldsymbol{\nu} \sigma(n))\sigma(v)$.

  Note that $\sigma(\text{И} z.A) \equiv_\alpha \text{И} y.\sigma(A\{z \leftarrow y\})$ for some $y \notin \mathit{afv}(m, x) \cup \mathit{fv}(A)$. Since $\mathit{ft}(\text{И} z.A) = \mathit{ft}(\text{И} y.A\{z \leftarrow y\})$, we also have $n \#_S \text{И} y.A\{z \leftarrow y\}$. Hence, by Lemma 7.4, $\sigma(n) \#_{\sigma(S)} \sigma(\text{И} y.A\{z \leftarrow y\})$. By (ИR) and (α), we can build a derivation $\langle\sigma(S)\rangle\, \sigma(\Gamma) \vdash \sigma(u) : \sigma(\text{И} z.A), \sigma(\Delta)$, since $\sigma(A\{z \leftarrow y\})\{y \leftarrow \sigma(n)\} \equiv_\alpha \sigma(A\{z \leftarrow n\})$.

- **(Case of (∀L))** We have a derivation of $\langle S\rangle\, \Gamma, u : \forall z.A \vdash \Delta$ concluded from a derivation of $\langle S\rangle\, \Gamma, u : A\{z \leftarrow n\} \vdash \Delta$. By induction hypothesis, we have

  $$\langle\sigma(S)\rangle\, \sigma(\Gamma), \sigma(u) : \sigma(A\{z \leftarrow n\}) \vdash \sigma(\Delta)$$

  By (α) and (∀L), we conclude $\langle\sigma(S)\rangle\, \sigma(\Gamma), \sigma(u) : \sigma(\forall z.A) \vdash \sigma(\Delta)$, since we have that $\sigma(\forall z.A) \equiv_\alpha \forall y.\sigma(A\{z \leftarrow y\})$ for some $y \in \mathit{afv}(m, x) \cup \mathit{fv}(A)$ and we can verify that $\sigma(A\{z \leftarrow y\})\{y \leftarrow \sigma(n)\} \equiv_\alpha \sigma(A\{z \leftarrow n\})$.

- **(Case of (ИL))** Similar to (ИR). ■

7.4 Cut Elimination

We present here the proof of cut elimination for the first-order fragment of the spatial logic. The proof relies on a few (size-preserving) proof transformations. We start by introducing some useful concepts.

Recall that a formula is *atomic* if it is not built from a logical connective, in our case, if it is either a propositional variable X or a message $n\langle m\rangle$. Now, we call *atomic* to a proof in which all instances of (Id) apply to atomic formulas. We call *simple* to an atomic proof in which all instances of the (τ) rule are simple. Hence, according to Definition 4.1, in a simple proof no occurrence of the (τ) rule in weighted. As discussed in Section 4.3, we give a size preserving transformation for building a "locally equivariant" derivation for any atomic derivation:

Lemma 4.4 [Closure] *Let ρ be a permutation of name terms n, each verifying $n \#_S fpv(A)$ and assume $u \doteq_S \rho t$ and $A' \equiv_S \rho{\cdot}A$. Then the following proof principles are admissible for the first-order fragment of the logic*

$$\frac{\vdash_n \langle S\rangle\, \Gamma, t : A \vdash \Delta}{\vdash_n \langle S\rangle\, \Gamma, u : A' \vdash \Delta}\ \text{(ClosL)} \qquad \frac{\vdash_n \langle S\rangle\, \Gamma \vdash t : A, \Delta}{\vdash_n \langle S\rangle\, \Gamma \vdash u : A', \Delta}\ \text{(ClosR)}$$

Moreover, if the original derivations are atomic and cut-free the resulting ones are simple and cut-free.

Proof. The proof proceeds by mutual induction on the size of the derivations (1) $\vdash_n \langle S\rangle\, \Gamma \vdash t : A, \Delta$ and (2) $\vdash_n \langle S\rangle\, \Gamma, t : A \vdash \Delta$.

(Case of (1)). Assume $\vdash_n \langle S\rangle\, \Gamma \vdash \Delta, t : A$. Possible ways of deriving this sequent are: (1) the last rule is either (Cut) or a logical rule (other than (τ)) acting on a principal formula in Δ or Γ, (2) the last rule is a structural rule acting on S or (И), (3) the last rule is (τ), or (4) the last rule is (Id) or a logical right rule acting on the principal formula $t : A$.

(Subcase 1) The result follows directly from the inductive hypothesis, possibly using (Ren) in the ($\forall$R) case.

(Subcase 2) If the last rule is some structural (S$-$) rule, the result is an immediate consequence of the induction hypothesis. If the rule is (И), we consider the sequent $\langle S\rangle\, \Gamma \vdash t : A, \Delta$ concluded from $\vdash_{n-1}$ $\langle S, v \doteq (\boldsymbol{\nu} x)\mathcal{X}, x \# N\rangle\, \Gamma \vdash t : A, \Delta$. By (Ren), there is a derivation $\vdash_{n-1} \langle S, v \doteq (\boldsymbol{\nu} y)\mathcal{X}, y \# N\rangle\, \Gamma \vdash t : A, \Delta$, where y is chosen not free neither in the original sequent, nor in $t' : A'$. By induction hypothesis, we conclude $\vdash_{n-1} \langle S, v \doteq (\boldsymbol{\nu} y)\mathcal{X}, y \# N\rangle\, \Gamma \vdash u : A', \Delta$. By (И), $\langle S\rangle\, \Gamma \vdash u : A', \Delta$ is obtained.

(Subcase 3) This is a special case of (Case 2, Subcase 2) below (since there is no right (τ) rule).

We now address (Subcase 4).

- **(Case of (Id))** Note that if $\langle S\rangle\, \Gamma \vdash t : A, \Delta$ is an instance of (Id), then either $\langle S\rangle\, \Gamma \vdash \Delta$ also is an instance of (Id) (and we can conclude by (W)), or Γ has the form $\Gamma', t' : B$ with $t \doteq_S t'$ and $A \equiv_S B$. In the latter case, note that $\langle S\rangle\, \Gamma', \rho t' : \rho{\cdot}B \vdash u : A', \Delta$ is an instance of (Id), since $u \equiv_S \rho t \equiv_S \rho t'$ and $A' \equiv_S \rho{\cdot}B$. Then, by a sequence of simple applications of (τ), we conclude $\vdash_1 \langle S\rangle\, \Gamma \vdash u : A', \Delta$.
- **(Case of (CR))** We have $\langle S\rangle\, \Gamma \vdash \Delta, t : A$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash \Delta, t : A, t : A$. By induction hypothesis twice, we have $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash \Delta, u : A', u : A'$. We then conclude by (CR).
- **(Case of ($\wedge$R))** We have $A = B \wedge C$ and $\langle S\rangle\, \Gamma \vdash \Delta, t : A$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash \Delta, t : B$ and $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash \Delta, t : C$. We have $A' = B' \wedge C'$ with $B' \equiv_S \rho{\cdot}B$ and $C' \equiv_S \rho{\cdot}C$. By induction hypothesis, we have $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash \Delta, u : B'$ and $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash \Delta, u : C'$. By ($\wedge$R), we conclude $\vdash_n \langle S\rangle\, \Gamma \vdash \Delta, u : A'$.
- **(Case of ($\Rightarrow$R))** We have $A = B \Rightarrow C$ and $\langle S\rangle\, \Gamma \vdash \Delta, t : A$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, t : B \vdash \Delta, t : C$. We have $A' = B' \Rightarrow C'$ with $B' \equiv_S \rho{\cdot}B$ and $C' \equiv_S \rho{\cdot}C$. By induction hypothesis (twice), we have $\vdash_{n-1} \langle S\rangle\, \Gamma, u : B' \vdash \Delta, u : C'$. By ($\Rightarrow$R), we conclude $\vdash_n \langle S\rangle\, \Gamma \vdash \Delta, u : A'$.
- **(Case of ($|$R))** We have $A = B|C$ and $\langle S\rangle\, \Gamma \vdash \Delta, t : A$ concluded from $\langle S\rangle\, \Gamma \vdash \Delta, u' : B$ and $\langle S\rangle\, \Gamma \vdash \Delta, u'' : C$, where $u'|u'' \doteq_S t$. We have $A' = B'|C'$ with $B' \equiv_S \rho{\cdot}B$ and $C' \equiv_S \rho{\cdot}C$ and $u \doteq_S \rho t \doteq_S \rho u'|\rho u''$. By induction hypothesis, we have $\langle S\rangle\, \Gamma \vdash \Delta, \rho u' : B'$ and $\langle S\rangle\, \Gamma \vdash \Delta, \rho u'' : C'$. By ($|$R), we conclude $\vdash_n \langle S\rangle\, \Gamma \vdash \Delta, u : A'$.
- **(Case of ($\triangleright$R))** We have $A = B \triangleright C$ and $\langle S\rangle\, \Gamma \vdash \Delta, t : A$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, \mathcal{X} : B \vdash \Delta, u' : C$ and $u' \doteq_S \mathcal{X}|t$. We have $A' = B' \triangleright C'$ with $B' \equiv_S \rho{\cdot}B$ and $C' \equiv_S \rho{\cdot}C$. By induction hypothesis we have $\vdash_{n-1} \langle S\rangle\, \Gamma, \rho\mathcal{X} : B' \vdash \Delta, \rho u' : C'$. By (Inst$\mathcal{I}$) with $\{\mathcal{X}{\leftarrow}\rho\mathcal{X}\}$ and induction

hypothesis, we have $\vdash_{n-1} \langle S\rangle\, \Gamma, \mathcal{X} : B' \vdash \Delta, \mathcal{X}|u : C'$, since $(\rho u')\{\mathcal{X}\leftarrow\rho\mathcal{X}\} \doteq_S \mathcal{X}|u$, because $u \doteq_S \rho t$ by assumption. By ($\triangleright$R), we get $\vdash_n \langle S\rangle\, \Gamma \vdash \Delta, u : A'$.

- **(Case of ($\Diamond$R))** We have $A = \Diamond B$ and $\langle S\rangle\, \Gamma \vdash \Delta, t : A$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash \Delta, v : B$ and $t \to_S v$. By induction hypothesis, $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash \Delta, \rho v : \rho{\cdot}B$. By ($\Diamond$R), we conclude $\vdash_n \langle S\rangle\, \Gamma \vdash \Delta, u : A'$, since $u = \rho t \to_S \rho v$ by (Swap Red).
- **(Case of (®R))** We have $A = q®B$ and $\langle S\rangle\, \Gamma \vdash \Delta, t : A$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash \Delta, u' : B$ and $t \doteq_S (\boldsymbol{\nu} q)u'$. We have $A' = (\rho q)®B'$ with $B' \equiv_N \rho{\cdot}B$. By induction hypothesis, $\vdash_{n-1}$ $\langle S\rangle\, \Gamma \vdash \Delta, \rho u' : B'$. By $(®R)$, we conclude $\vdash_n \langle S\rangle\, \Gamma \vdash \Delta, u : A'$, since $(\boldsymbol{\nu}\rho q)\rho u' \doteq_S u \doteq_S \rho t$.
- **(Case of ($\oslash$R))** Similar to (®R).
- **(Case of (0R))** Immediate.
- **(Case of (ИR))** We have $A = \text{И}x.B$ and $\langle S\rangle\, \Gamma \vdash t : A, \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash t : B\{x\leftarrow p\}, \Delta$ where $p \,\#_S\, \text{И}x.B$, and $t \doteq_S (\boldsymbol{\nu} p)v$.

  By (α) we can assume that x is not free in ρ so that $\rho{\cdot}A \equiv_\alpha \text{И}x.\rho{\cdot}B_{\{x\}}$ and thus $A' \equiv_S \text{И}x.\rho{\cdot}B_{\{x\}}$. We have $\rho{\cdot}B\{x\leftarrow p\} = (\rho{\cdot}B_{\{x\}})\{x\leftarrow \rho p\}$. Write $q \triangleq \rho p$.

  Hence, by induction hypothesis, we have $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash u : (\rho{\cdot}B_{\{x\}})\{x\leftarrow q\}, \Delta$.

  By (ИR), we conclude $\vdash_n \langle S\rangle\, \Gamma \vdash \Delta, u : A'$, since by Lemma 7.4 $q \,\#_S\, A'$, and $\rho t \doteq_S u \doteq_S (\boldsymbol{\nu} q)\rho v$.
- **(Case of ($\forall$R))** We have $A = \forall x.B$ and $\langle S\rangle\, \Gamma \vdash t : A, \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash t : B\{x\leftarrow y\}, \Delta$, where y is not free in the conclusion. By (α) we can assume that x does not occur in ρ, so that $\rho{\cdot}A \equiv_\alpha \forall x.\rho{\cdot}B_{\{x\}}$, and then $A' \equiv_S \rho{\cdot}A \equiv_\alpha \forall x.\rho{\cdot}B_{\{x\}}$.

  By (Inst$\mathcal{N}$) we have $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash t : B\{x\leftarrow \rho^{-1}y\}, \Delta$.

  We have $\rho{\cdot}B_{\{x\}}\{x\leftarrow y\} = \rho{\cdot}B_{\{y\}} \equiv_S \rho{\cdot}B\{x\leftarrow \rho^{-1}y\}$.

  By induction hypothesis, we have $\vdash_{n-1} \langle S\rangle\, \Gamma \vdash u : \rho{\cdot}B_{\{x\}}\{x\leftarrow y\}, \Delta$. By ($\forall$R) we conclude $\vdash_n \langle S\rangle\, \Gamma \vdash u : A', \Delta$.

(Case of (2)) Assume $\vdash_n \langle S\rangle\, \Gamma, t : A \vdash \Delta$. Like in Case (1) above, possible ways of deriving this sequent are: (1) the last rule is either (Cut) or a logical rule (other than (τ)) acting on a principal formula in Δ or Γ, (2) the last rule is a structural rule acting on S or (И), (3) the last rule is (τ), or (4) the last rule is (Id) or a logical left rule acting on the principal formula $t : A$.

Subcases (1) and (2) are treated as in (Case (1)).

(Subcase 3) Assume first that the given instance of (τ) is not simple.

If the rule applies to the distinguished occurrence $t : A$, the sequent $\vdash_n \langle S\rangle\, \Gamma, t : A \vdash \Delta$ is concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, (m \leftrightarrow n)t : \{m{\leftrightarrow}n\}{\cdot}A \vdash \Delta$ for some m, n such that $m, n \,\#_S\, \mathit{fpv}(A)$. By applying the induction hypothesis with the permutation $\rho' = \rho(m \leftrightarrow n)$, we get a simple derivation $\vdash_{n-1} \langle S\rangle\, \Gamma, u : A' \vdash \Delta$.

Otherwise, the sequent $\langle S\rangle\, \Gamma, t : A \vdash \Delta$ results from $\vdash_{n-1} \langle S\rangle\, \Gamma', (m \leftrightarrow n)v : \{m{\leftrightarrow}n\}{\cdot}B, t : A \vdash \Delta$, where $\Gamma = \Gamma', v : B$. By induction hypothesis, there is a simple derivation $\vdash_{n-1} \langle S\rangle\, \Gamma', v : B, t : A \vdash \Delta$ (we pick the permutation $(m \leftrightarrow n)$). By induction hypothesis again (twice), we obtain a simple derivation $\vdash_{n-1} \langle S\rangle\, \Gamma, u : A' \vdash \Delta$. The case where ($\tau$) is applied in Δ is dealt in the same way.

If the instance of (τ) is simple, then we immediately obtain $\vdash_1 \langle S\rangle\, \Gamma, u : A' \vdash \Delta$, by a sequence of simple applications of (τ).

We now consider (Subcase 4)

- **(Case of (Id))** Note that if $\langle S\rangle\, \Gamma, t : A \vdash \Delta$ is an instance of (Id), then either $\langle S\rangle\, \Gamma \vdash \Delta$ also is (and thus we conclude by (W)), or Δ has the form $t' : B, \Delta'$ with $t' \doteq_S t$ and $B \equiv_S A$. In the latter case, note that $\langle S\rangle\, \rho u : \rho{\cdot}A', \Gamma \Vdash \Delta$ is also an instance of (Id). By a sequence of simple applications of (τ), we conclude $\langle S\rangle\, u : A', \Gamma \Vdash \Delta$.

- **(Case of (CL))** Similar to (CR).
- **(Case of ($\wedge$L) and ($\Rightarrow$L))** Similar to ($\wedge$R) and ($\Rightarrow$R).
- **(Case of ($|$L))** We have $A = B|C$ where $\langle S\rangle\, \Gamma, t : A \Vdash \Delta$ results from $\vdash_{n-1} \langle S, t \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : B, \mathcal{Y} : C \Vdash \Delta$. We have $A' = B'|C'$ where $B' \equiv_S \rho{\cdot}B$ and $C' \equiv_S \rho{\cdot}C$.

  By (Inst$\mathcal{I}$) with $\{\mathcal{X}{\leftarrow}\rho\mathcal{X}\}$ and $\{\mathcal{Y}{\leftarrow}\rho\mathcal{Y}\}$ we obtain $\vdash_{n-1} \langle S, t \doteq \rho\mathcal{X}|\rho\mathcal{Y}\rangle\, \Gamma, \rho\mathcal{X} : B, \rho\mathcal{Y} : C \Vdash \Delta$. By induction hypothesis we get $\vdash_{n-1} \langle S, t \doteq \rho\mathcal{X}|\rho\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : B', \mathcal{Y} : C' \Vdash \Delta$. But then $\vdash_{n-1} \langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : B', \mathcal{Y} : C' \Vdash \Delta$ by (W) and (CS). By ($|$L), we conclude $\vdash_n \langle S\rangle\, \Gamma, u : A' \Vdash \Delta$.
- **(Case of ($\triangleright$L))** We have $A = B \triangleright C$ and $\langle S\rangle\, \Gamma, t : A \Vdash \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash \Delta, v : B$ and $\vdash_{n-1} \langle S\rangle\, \Gamma, v|t : C \Vdash \Delta$. We have $A' = B' \triangleright C'$ where $B' \equiv_S \rho{\cdot}B$ and $C' \equiv_S \rho{\cdot}C$. By induction hypothesis, we have $\vdash_{n-1} \langle S\rangle\, \Gamma, \rho v|\rho t : C' \Vdash \Delta$ and $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash \rho v : B', \Delta$. By ($\triangleright$L), we conclude $\vdash_n \langle S\rangle\, \Gamma, u : A' \Vdash \Delta$.
- **(Case of ($\Diamond$L))** We have $A = \Diamond B$ and $\langle S\rangle\, \Gamma, t : A \Vdash \Delta$ concluded from $\vdash_{n-1} \langle S, t \to \mathcal{X}\rangle\, \Gamma, \mathcal{X} : B \Vdash \Delta$. By (Inst$\mathcal{I}$) with $\{X{\leftarrow}\rho X\}$ we get a derivation $\vdash_{n-1} \langle S, t \to \rho\mathcal{X}\rangle\, \Gamma, \rho\mathcal{X} : B \Vdash \Delta$. By induction hypothesis $\vdash_{n-1} \langle S, t \to \rho\mathcal{X}\rangle\, \Gamma, \mathcal{X} : \rho{\cdot}B \Vdash \Delta$. By (W) $\vdash_{n-1} \langle S, t \to \rho\mathcal{X}, u \to \mathcal{X}\rangle\, \Gamma, \mathcal{X} : \rho{\cdot}B \Vdash \Delta$. By (CS), $\vdash_{n-1} \langle S, u \to X\rangle\, \Gamma, \mathcal{X} : \rho{\cdot}B \Vdash \Delta$. By ($\Diamond$L), we conclude $\vdash_n \langle S\rangle\, \Gamma, u : A' \Vdash \Delta$.
- **(Case of (®L), ($\oslash$L) and (0L))** Similar to ($|$L).
- **(Case (ИL))** We have $A = \text{И}x.B$ and $\langle S\rangle\, \Gamma, t : A \Vdash \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, t : B\{x{\leftarrow}p\} \Vdash \Delta$, where $t \doteq_S (\boldsymbol{\nu} p)v$ and $p \#_S \text{И}x.B$.

  Like in case for (ИL) above, by (α) we can assume that x is not free in ρ so that $A' \equiv_S \text{И}x.\rho{\cdot}B_{\{x\}}$, and we have $\rho{\cdot}B\{x{\leftarrow}p\} = (\rho{\cdot}B_{\{x\}})\{x{\leftarrow}\rho p\}$. Write $q \triangleq \rho p$.

  By induction hypothesis, we conclude $\vdash_{n-1} \langle S\rangle\, \Gamma, u : (\rho{\cdot}B_{\{x\}})\{y{\leftarrow}q\} \Vdash \Delta$. By ($\text{И}$L), we conclude $\vdash_n \langle S\rangle\, \Gamma, u : A' \Vdash \Delta$, since $q \#_S A'$ and $u \doteq_S (\boldsymbol{\nu} q)\rho v$.
- **(Case of ($\forall$L))** We have $A = \forall x.B$ and $\langle S\rangle\, \Gamma, t : A \Vdash \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, t : B\{x{\leftarrow}m\} \Vdash \Delta$. Like in the case for ($\text{И}$L) above, we may assume x is not free in ρ so that $A' \equiv_S \forall x.\rho{\cdot}B_{\{x\}}$, and we have $\rho{\cdot}B\{x{\leftarrow}m\} = (\rho{\cdot}B_{\{x\}})\{x{\leftarrow}\rho m\}$. By induction hypothesis, we have $\vdash_{n-1} \langle S\rangle\, \Gamma, u : (\rho{\cdot}B_{\{x\}})\{x{\leftarrow}\rho m\} \Vdash \Delta$. By ($\forall$L), we get the conclusion $\vdash_n \langle S\rangle\, \Gamma, u : A' \Vdash \Delta$. ∎

From now on, we restrict ourselves to the first-order fragment of our logic, and introduce an alternative (but equivalent, see Proposition 7.11 below) contraction-free proof system CF. As usual, the motivation for introducing a contraction-free variant is basically to bring some simplification to the proof of cut elimination, and for proof-search purposes. CF is obtained from the basic system of Section 3 by removing the contraction rules (CL) and (CR), and replacing certain logical rules by the variants shown in Figure 13. The CF rules are identical to the basic ones, except in that they absorb a contraction step (*cf.* the system **G3c** in [15]).

The replaced rules are precisely the non-invertible ones: ($\forall$L), ($|$R), ($\triangleright$L), (ИL), (ИR), (®R), and ($\Diamond$R). Note that in classical logic (e.g. Gentzen's LK) ($\forall$L) is not invertible, and in classical linear logic ($\otimes$R) is not invertible (*cf.*, ($|$R)) and ($\multimap$L) is not invertible (*cf.* ($\triangleright$)). We also substitute the general (Id) axiom by the atomic (AId).

Lemma 7.6 *If a sequent has a derivation in* CF*, it has a derivation in the basic system.*

$$\frac{\langle S\rangle\,\Gamma \vdash v : A, u : A|B, \Delta \quad \langle S\rangle\,\Gamma \vdash t : B, u : A|B, \Delta \;\; u \doteq_S v|t}{\langle S\rangle\,\Gamma \vdash u : A|B, \Delta}\ (|\mathrm{RK}) \qquad \frac{\langle S\rangle\,\Gamma, u : A \triangleright B \vdash_N t : A, \Delta \quad \langle S\rangle\,\Gamma, u : A \triangleright B, t|u : B \vdash_N \Delta}{\langle S\rangle\,\Gamma, u : A \triangleright B \vdash_N \Delta}\ (\triangleright\mathrm{LK})$$

$$\frac{u \doteq_S (\boldsymbol{\nu} n)t \quad \langle S\rangle\,\Gamma \vdash t : A, u : n®A, \Delta}{\langle S\rangle\,\Gamma \vdash u : n®A, \Delta}\ (®\mathrm{RK}) \qquad \frac{[\ A \textit{ is an atomic formula }]\quad u \doteq_S u' \;\; A \equiv_S A'}{\langle S\rangle\,\Gamma, u : A \vdash u' : A', \Delta}\ (\mathrm{AId})$$

$$\frac{\langle S\rangle\,\Gamma \vdash t : A, u : \Diamond A, \Delta \;\; u \rightarrow_S t}{\langle S\rangle\,\Gamma \vdash u : \Diamond A, \Delta}\ (\Diamond\mathrm{RK}) \qquad \frac{\langle S\rangle\,\Gamma, u : \forall x.A, u : A\{x\leftarrow m\} \vdash \Delta}{\langle S\rangle\,\Gamma, u : \forall x.A \vdash \Delta}\ (\forall\mathrm{LK})$$

$$\frac{u \doteq_S (\boldsymbol{\nu} n)v \quad n \#_S \text{И} x.A \quad \langle S\rangle\,\Gamma \vdash u : \text{И} x.A, u : A\{x\leftarrow n\}, \Delta}{\langle S\rangle\,\Gamma \vdash u : \text{И} x.A, \Delta}\ (\text{И}\mathrm{R}) \qquad \frac{u \doteq_S (\boldsymbol{\nu} n)v \quad n \#_S \text{И} x.A \quad \langle S\rangle\,\Gamma, u : \text{И} x.A, u : A\{x\leftarrow n\} \vdash \Delta}{\langle S\rangle\,\Gamma, u : \text{И} x.A \vdash \Delta}\ (\text{И}\mathrm{L})$$

Figure 13: Contraction Free Rules

Proof. Any derivation in CF can be immediately transformed into a derivation in the basic system, since each CF rule not belonging to the basic system can be simulated by the corresponding basic rule followed by contraction. ■

Moreover, since the proof transformations given in Lemmas 4.2 and 4.4 are completely structure-preserving, we can verify that

Lemma 7.7 (Admissible Rules for the CF system) *The size preserving admissible proof principles in Lemma 4.2 and Lemma 4.4 hold exactly as stated for the contraction free system.*

Lemma 7.8 (Inversion) *The following size-preserving proof principles are admissible:*

1. *If* $\vdash_n \langle S\rangle\,\Gamma, u : A \wedge B \vdash \Delta$ *then* $\vdash_n \langle S\rangle\,\Gamma, u : A, u : B \vdash \Delta$.
2. *If* $\vdash_n \langle S\rangle\,\Gamma \vdash u : A \wedge B, \Delta$ *then* $\vdash_n \langle S\rangle\,\Gamma \vdash u : A, \Delta$ *and* $\vdash_n \langle S\rangle\,\Gamma \vdash u : B, \Delta$.
3. *If* $\vdash_n \langle S\rangle\,\Gamma \vdash u : A \Rightarrow B, \Delta$ *then* $\langle S\rangle\,\Gamma, u : A \vdash u : B, \Delta$.
4. *If* $\vdash_n \langle S\rangle\,\Gamma, u : A \Rightarrow B \vdash \Delta$ *then* $\vdash_n \langle S\rangle\,\Gamma, u : B \vdash \Delta$ *and* $\vdash_n \langle S\rangle\,\Gamma \vdash u : A, \Delta$.
5. *If* $\vdash_n \langle S\rangle\,\Gamma \vdash u : \forall x.A, \Delta$ *then* $\vdash_n \langle S\rangle\,\Gamma \vdash u : A\{x\leftarrow y\}, \Delta$*, for any* y *not free in the first sequent.*
6. *If* $\vdash_n \langle S\rangle\,\Gamma \vdash u : A \triangleright B, \Delta$ *then* $\vdash_n \langle S\rangle\,\Gamma, \mathcal{X} : A \vdash \mathcal{X}|u : B, \Delta$*, for any* $\mathcal{X}$ *not free in the first sequent.*
7. *If* $\vdash_n \langle S\rangle\,\Gamma, u : A|B \vdash \Delta$ *then* $\vdash_n \langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\,\Gamma, \mathcal{X} : A, \mathcal{Y} : B \vdash \Delta$*, for any* $\mathcal{X}, \mathcal{Y}$ *not free in the first sequent.*
8. *If* $\vdash_n \langle S\rangle\,\Gamma, u : n®A \vdash \Delta$ *then* $\vdash_n \langle S, u \doteq (\boldsymbol{\nu} n)\mathcal{X}\rangle\,\Gamma, \mathcal{X} : A, \vdash \Delta$*, for any* $\mathcal{X}$ *not free in the first sequent.*
9. *If* $\vdash_n \langle S\rangle\,\Gamma, u : \mathbf{0} \vdash \Delta$ *then* $\vdash_n \langle S, u \doteq \mathbf{0}\rangle\,\Gamma \vdash \Delta$.

Moreover, if the original derivations are cut-free, so are the resulting ones.

Proof. By induction on the size of the proof of the given sequents. We present a detailed argument for (7), the remaining cases are handled in a similar way. We consider three subcases:

(a) If the last step is an application of (|L) to the distinguished formula $u : A|B$, there is nothing to prove.

(b) The last step in an application of (τ) to the distinguished formula $u : A|B$. Hence we have $\vdash_{n-1} \langle S\rangle\, \Gamma, (\tau)u : \tau{\cdot}A|\tau{\cdot}B \Vdash \Delta$. By induction hypothesis, we have $\vdash_{n-1} \langle S, \tau u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : \tau{\cdot}A, \mathcal{Y} : \tau{\cdot}B \Vdash \Delta$. By (Inst$\mathcal{I}$) we have $\vdash_{n-1} \langle S, \tau u \doteq \tau\mathcal{X}|\tau\mathcal{Y}\rangle\, \Gamma, \tau\mathcal{X} : \tau{\cdot}A, \tau\mathcal{Y} : \tau{\cdot}B \Vdash \Delta$. By (ClosL), (W) and (CS), we have $\vdash_{n-1} \langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : A, \mathcal{Y} : B \Vdash \Delta$.

(c) Otherwise, the sequent $\langle S\rangle\, \Gamma, u : A|B \Vdash \Delta$ is concluded from $k = 1$ or $k = 2$ premises of the form $\langle S_i\rangle\, \Gamma_i, u : A|B \Vdash \Delta_i$, for $i = 1, \ldots, k$ and possibly some premises of the form $S \vdash c$, by an application of some inference rule $(\mathcal{R})$ acting either on a principal formula in Γ or Δ, or in $\langle S\rangle$. By induction hypothesis, we conclude $\vdash_{n-1} \langle S_i, u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma_i, \mathcal{X} : A, \mathcal{Y} : B \Vdash \Delta_i$ for $i = 1, \ldots, k$, where $\mathcal{X}$ and $\mathcal{Y}$ can be chosen fresh with respect to S, S_i, Γ, Δ, Γ_i and Δ_i (so that any eigenvariable condition required for the application of $(\mathcal{R})$ still holds). By $(\mathcal{R})$, we conclude $\vdash_n S, u \doteq \mathcal{X}|\mathcal{Y}; \Gamma, u : A, u : B \Vdash \Delta$. ■

Lemma 7.9 (Admissibility of contraction) *In the contraction-free system the following size-preserving proof principles are admissible*

$$\frac{\vdash_n \langle S\rangle\, \Gamma \Vdash u : A, u : A, \Delta}{\vdash_n \langle S\rangle\, \Gamma, u : A \Vdash \Delta}\ \text{(CR)} \qquad \frac{\vdash_n \langle S\rangle\, \Gamma, u : A, u : A \Vdash \Delta}{\vdash_n \langle S\rangle\, \Gamma, u : A \Vdash \Delta}\ \text{(CL)}$$

Moreover, if the original derivations are cut-free and simple, so are the resulting ones.

Proof. The properties (CR) and (CR) are proved by mutual induction on the size of the respective derivations. If the last rule of the derivation is a (S$-$) rule, (Cut), or a logical other than (Id), applying to some formula in Γ or Δ, then the result follows directly from the induction hypothesis. If last rule is (Id), identifying atomic formulas in Γ or Δ, then $\langle S\rangle\, \Gamma \Vdash \Delta$ is an instance of (Id). The conclusion can then be obtained by adding the required formulas to the left and right context of this sequent. Otherwise, we consider the case of each possible rule acting on one of the distinguished occurrences of $u : A$ (and then note that such rule cannot be an (Id) axiom).

(Case of (CR))

- **(Case of (Id))** Immediate, since just one of the $u : A$ can be relevant to (Id).
- **(Case of (F R))** We have $A = \mathtt{F}$ and $\langle S\rangle\, \Gamma \Vdash u : \mathtt{F}, u : \mathtt{F}, \Delta$, concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash u : \mathtt{F}, \Delta$, which is just what we need.
- **(Case of (∧R))** We have $A = B \wedge C$ and $\vdash_n \langle S\rangle\, \Gamma \Vdash u : A, u : A, \Delta$, concluded from $\vdash_k \langle S\rangle\, \Gamma \Vdash u : B, u : A, \Delta$ and $\vdash_{k'} \langle S\rangle\, \Gamma \Vdash u : C, u : A, \Delta$ where $n = 1 + k + k'$. By Lemma 7.8(2), we have $\vdash_k \langle S\rangle\, \Gamma \Vdash u : B, u : B, \Delta$ and $\vdash_{k'} \langle S\rangle\, \Gamma \Vdash u : C, u : C, \Delta$. By induction hypothesis, we have $\vdash_k \langle S\rangle\, \Gamma \Vdash u : B, \Delta$ and $\vdash_{k'} \langle S\rangle\, \Gamma \Vdash u : C, \Delta$. We get the conclusion by (∀R).
- **(Case of (⇒R))** We have $A = B \Rightarrow C$ and $\langle S\rangle\, \Gamma \Vdash u : A, u : A, \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, u : B \Vdash u : C, u : A, \Delta$. We conclude by Lemma 7.8(3).
- **(Case of (|RK))** We have $A = B|C$ and $\langle S\rangle\, \Gamma \Vdash u : A, u : A, \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash v : B, u : A, u : A, \Delta$ and $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash t : C, u : A, u : A, \Delta$ and $v|t \doteq_S u$. By induction hypothesis, we have $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash v : B, u : A, \Delta$ and $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash t : C, u : A, \Delta$. We then get the conclusion by (|RK).
- **(Case of (▷R))** We have $A = B \triangleright C$ and $\vdash_n \langle S\rangle\, \Gamma \Vdash u : A, u : A, \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, \mathcal{X} : B \Vdash v : C, u : A, \Delta$ and $v \doteq_S \mathcal{X}|u$. By Lemma 7.8(6), we have $\vdash_{n-1} \langle S\rangle\, \Gamma, \mathcal{X} : B, \mathcal{Y} : B \Vdash v : C, \mathcal{Y}|u : C, \Delta$ for some fresh $\mathcal{Y}$. By (Inst$\mathcal{I}$) we get $\vdash_{n-1} \langle S\rangle\, \Gamma, \mathcal{X} : B, \mathcal{X} : B \Vdash v : C, \mathcal{X}|u : C, \Delta$. By (ClosR) with the identity permutation, we have $\vdash_{n-1} \langle S\rangle\, \Gamma, \mathcal{X} : B, \mathcal{X} : B \Vdash v : C, v : C, \Delta$, since $v \doteq_S \mathcal{X}|u$. By induction hypothesis, we get then $\vdash_{n-1} \langle S\rangle\, \Gamma, \mathcal{X} : B \Vdash v : C, \Delta$. Then, the conclusion is obtained by (▷R).

- **(Cases of (®RK))** Similar to (|RK).
- **(Cases of (◊RK))** Similar to (|RK).
- **(Case of (⊘RK))** Similar to (®RK).
- **(Case of (0R))** Immediate.
- **(Case of (∀R))** We have $A = \forall x.B$ and $\langle S\rangle\, \Gamma \Vdash u : A, u : A, \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash u : B\{x{\leftarrow}y\}, u : A, \Delta$, where y is not free in the conclusion. By Lemma 7.8(5), we have $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash u : B\{x{\leftarrow}y\}, u : B\{x{\leftarrow}z\}, \Delta$, where z is not free in the conclusion. By (Inst$\mathcal{N}$) with $\{z{\leftarrow}y\}$, we have $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash u : B\{x{\leftarrow}y\}, u : B\{x{\leftarrow}y\}, \Delta$. By the induction hypothesis, we get $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash u : B\{x{\leftarrow}y\}, \Delta$ and we conclude by (∀R).
- **(Case of (ИRK))** We have $A = \text{И}x.B$ and $\langle S\rangle\, \Gamma \Vdash u : A, u : A, \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash u : B\{x{\leftarrow}p\}, u : A, u : A, \Delta$ where $p \#_S \text{И}x.B$, and $t \doteq_S (\boldsymbol{\nu} p)v$. By induction hypothesis, we get $\vdash_{n-1} \langle S\rangle\, \Gamma \Vdash u : B\{x{\leftarrow}p\}, u : A, \Delta$, we then conclude by (ИRK).

(Case of (CL))

- **(Case of (Id))** Immediate, like for the right case above.
- **(Case of (FL))** We have $A = \mathtt{F}$ where $\vdash_1 \langle S\rangle\, \Gamma, u : \mathtt{F}, u : \mathtt{F} \Vdash \Delta$ is an instance of (FL). Then $\vdash_1 \langle S\rangle\, \Gamma, u : \mathtt{F} \Vdash \Delta$ also is.
- **(Case of (∧L) and (⇒L))** Similar to (∧R) and (⇒R).
- **(Case of (|L))** We have $A = B|C$ where $\langle S\rangle\, \Gamma, u : A, u : A \Vdash \Delta$ results from $\vdash_{n-1} \langle S, t \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : B, \mathcal{Y} : C, u : A \Vdash \Delta$. By Lemma 7.8(7), we have $\vdash_{n-1} \langle S, t \doteq \mathcal{X}|\mathcal{Y}, t \doteq \mathcal{Z}|\mathcal{W}\rangle\, \Gamma, \mathcal{X} : B, \mathcal{Y} : C, \mathcal{Z} : B, \mathcal{W} : C \Vdash \Delta$, where $\mathcal{Z}$ and $\mathcal{W}$ are chosen fresh. By (Inst$\mathcal{I}$) with $\{\mathcal{Z}{\leftarrow}\mathcal{X}\}$ and $\{\mathcal{W}{\leftarrow}\mathcal{Y}\}$ we get $\vdash_{n-1} \langle S, t \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : B, \mathcal{Y} : C, \mathcal{X} : B, \mathcal{Y} : C \Vdash \Delta$. By induction hypothesis and (CS), we obtain a derivation $\vdash_{n-1} \langle S, t \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : B, \mathcal{Y} : C \Vdash \Delta$. We then conclude by (|L).
- **(Case of (▷LK))** We have $A = B \triangleright C$ and $\langle S\rangle\, \Gamma, u : A, u : A \Vdash \Delta$ concluded from $\vdash_k \langle S\rangle\, \Gamma, u : A, u : A \Vdash \Delta, v : B$ and $\vdash_{k'} \langle S\rangle\, \Gamma, u : A, u : A, v|t : C \Vdash \Delta$. By induction hypothesis, we have $\vdash_k \langle S\rangle\, \Gamma, u : A \Vdash \Delta, v : B$ and $\vdash_{k'} \langle S\rangle\, \Gamma, u : A, v|t : C \Vdash \Delta$. We conclude $\vdash_n \langle S\rangle\, \Gamma, u : A \Vdash \Delta$ by (▷LK).
- **(Case of (®L), (⊘L)** Similar to (|L).
- **(Case of (0L))** We have $\langle S\rangle\, \Gamma, u : \mathbf{0}, u : \mathbf{0} \Vdash \Delta$ concluded from $\vdash_{n-1} \langle S, u \doteq \mathbf{0}\rangle\, \Gamma, u : \mathbf{0} \Vdash \Delta$. By Lemma 7.8, we have $\vdash_{n-1} \langle S, u \doteq \mathbf{0}\rangle\, \Gamma \Vdash \Delta$, we conclude by (0L).
- **(Case (ИLK))** We have $A = \text{И}x.B$ and $\langle S\rangle\, \Gamma, u : A, u : A \Vdash \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, u : B\{x{\leftarrow}p\}, u : A, u : A \Vdash \Delta$, where $t \doteq_S (\boldsymbol{\nu} p)v$ and $p \#_S \text{И}x.B$. By induction hypothesis, we get $\vdash_{n-1} \langle S\rangle\, \Gamma, u : B\{x{\leftarrow}p\}, u : A \Vdash \Delta$, we conclude by (ИLK).
- **(Case of (τ))** If this occurrence of (τ) is simple, then just one of the occurrences of $u : A$ can be relevant to the (Id) axiom that leads to it. Otherwise, we have $\langle S\rangle\, \Gamma, u : A, u : A \Vdash \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, (m \leftrightarrow n)u : \{m{\leftrightarrow}n\}{\cdot}A, u : A \Vdash \Delta$. By (ClosL), we get $\vdash_{n-1} \langle S\rangle\, \Gamma, (m \leftrightarrow n)u : \{m{\leftrightarrow}n\}{\cdot}A, (m \leftrightarrow n)u : \{m{\leftrightarrow}n\}{\cdot}A \Vdash \Delta$. By induction hypothesis, we have $\vdash_{n-1} \langle S\rangle\, \Gamma, (m \leftrightarrow n)u : \{m{\leftrightarrow}n\}{\cdot}A \Vdash \Delta$. By (τ), we conclude $\vdash_n \langle S\rangle\, \Gamma, u : A \Vdash \Delta$.
- **(Case of (∀LK))** We have $A = \forall x.B$ and $\langle S\rangle\, \Gamma, u : A, u : A \Vdash \Delta$ concluded from $\vdash_{n-1} \langle S\rangle\, \Gamma, u : B\{x{\leftarrow}m\}, u : A, u : A \Vdash \Delta$. By induction hypothesis, we get $\vdash_{n-1} \langle S\rangle\, \Gamma, u : B\{x{\leftarrow}m\}, u : A \Vdash \Delta$, we conclude by (∀LK). ■

Lemma 7.10 *If a sequent is an instance of* (Id)*, it has a cut-free* CF *derivation.*

Proof. We have $\langle S\rangle\, \Gamma, u : B \vdash v : A, \Delta$, where $A \equiv_S B$ and $u \doteq_S v$. By induction on the structure of the formula B we show that this sequent has an atomic derivation: in the base case the sequent is itself an instance of (AId). We show a few cases

- **(Case of $A = A_1|A_2$)** We have $A \equiv_S B$ hence $B = B_1|B_2$ with $A_1 \equiv_S B_1$ and $A_2 \equiv_S B_2$ for some formulas B_1 and B_2. By induction hypothesis, there are CF proofs of $\langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : B_1, \mathcal{Y} : B_2 \vdash \mathcal{X} : A_1, v : A, \Delta$ and $\langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : B_1, \mathcal{Y} : B_2 \vdash \mathcal{Y} : A_2, v : A, \Delta$. By an application of (|RK) we get $\langle S, u \doteq \mathcal{X}|\mathcal{Y}\rangle\, \Gamma, \mathcal{X} : B_1, \mathcal{Y} : B_2 \vdash v : A_1|A_2, \Delta$. We then conclude by (|L).
- **(Case of $A = A_1 \triangleright A_2$)** We have $A \equiv_S B$ hence $B = B_1 \triangleright B_2$ with $A_1 \equiv_S B_1$ and $A_2 \equiv_S B_2$ for some formulas B_1 and B_2. By induction hypothesis, there are CF proofs of $\langle S\rangle\, \Gamma, \mathcal{X} : A_1, u : B \vdash \mathcal{X} : B_1, \Delta$ and $\langle S\rangle\, \Gamma, \mathcal{X}|u : B_2, u : B \vdash \mathcal{X}|v : A_2, \Delta$. By an application of ($\triangleright$LK) we get $\langle S\rangle\, \Gamma, \mathcal{X} : A_1, u : B \vdash \mathcal{X}|v : A_2, \Delta$. We conclude by ($\triangleright$R).
- **(Case of $A = \forall x.C$)** We have $A \equiv_S B$, hence $B = \forall x.D$ with $C \equiv_S D$. Let y be chosen such that y does not occur free in the given sequent. Then $C\{x\leftarrow y\} \equiv_S D\{x\leftarrow y\}$. By induction hypothesis there is a CF proof of $\langle S\rangle\, \Gamma, u : B, u : C\{x\leftarrow y\} \vdash v : D\{x\leftarrow y\}, \Delta$. We conclude by ($\forall$LK) followed by ($\forall$R).
- **(Case of $A = \text{И}x.C$)** We have $B \equiv_S \text{И}x.D$, hence $C\{x\leftarrow y\} \equiv_S D\{x\leftarrow y\}$, for any name variable y that does not occur free in the given sequent. Let N be the set of all name and propositional variables occurring free in the given sequent. Let $\langle S'\rangle \triangleq \langle S, v \doteq (\boldsymbol{\nu} y)\mathcal{X}, y \# N\rangle$, where $\mathcal{X}$ and y are also chosen not free in the sequents under consideration. By induction hypothesis, there is a CF proof of $\langle S'\rangle\, \Gamma, u : B, u : C\{x\leftarrow y\} \vdash v : D\{x\leftarrow y\}, v : A, \Delta$. Note that $y \#_{S'} \text{И}x.C$, $y \#_{S'} \text{И}x.D$ and $u \doteq_{S'} (\boldsymbol{\nu} y)\mathcal{X} \doteq_{S'} v$. We conclude by (ИLK), (ИRK), and finally (И). ∎

Proposition 7.11 *A sequent is derivable in the basic system if and only if it is derivable in* CF.

Proof. Any basic derivation of a sequent can be transformed into a CF derivation by first eliminating all occurrences of (Id) which are not atomic using Lemma 7.10, and then iterate Lemma 7.9 for all occurrences of (CL) and (CR). ∎

Definition 7.12 (Single-cut derivation) *A* single-cut derivation *is a derivation with a single instance of the (Cut) rule, occurring at its root.*

Lemma 7.13 (Cut Lemma) *If a sequent has a single-cut simple* CF *derivation then it has a simple* CF *cut-free derivation.*

Proof. The root of the derivation of the given sequent has the form

$$\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\, \Gamma \vdash u : A, \Delta} \quad \dfrac{\pi_2(m)}{\langle S\rangle\, \Gamma, u : A \vdash \Delta}}{\langle S\rangle\, \Gamma \vdash \Delta}\ \text{(Cut)}$$

where $\pi_1(n)$ and $\pi_2(m)$ are cut-free simple derivations for the sequents $\langle S\rangle\, \Gamma \vdash u : A, \Delta$ and $\langle S\rangle\, \Gamma, u : A \vdash \Delta$, of sizes n and m respectively. We use the notation

$$\dfrac{\pi(n)}{\langle S\rangle\, \Gamma \vdash \Delta}$$

to assert that π is a derivation of size n of the sequent $\langle S\rangle\, \Gamma \vdash \Delta$. The proof proceeds by induction on the measure $(|A|, n+m)$, where $|A|$ is the structural complexity of the cut-formula A, $n+m$ is the sum of the sizes n and m of the derivations that occurs as premises of the cut, and the pairs $(|A|, m+n)$ are ordered lexicographically. We split the various possible forms of such premises as follows: (1) one of the premises is an instance of (Id) or (τ), (2) one of the premises is an instance of a structural rule, (3) one of the premises is an instance of ($\mathsf{И}$), (4) one of the premises is an instance of a logical rule that does not introduce the cut-formula, or (5) both premises are instances of logical rules, both introducing the cut formula.

(1) (Case (Cut) - (Id/τ)) Suppose (Id / τ) occurs in the right premise of the cut. Since the derivation is simple by assumption, it must have the form

$$\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\, \Gamma \vdash u : A, \Delta} \qquad \begin{array}{c}\langle S\rangle\, \Gamma^* \vdash \Delta \ \text{(Id)} \\ \vdots \\ \langle S\rangle\, \Gamma, u : A \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash \Delta}$$

where the dots stand for a sequence of $k \geq 0$ applications of the (τ) rule. Hence, Δ has the form $t : B, \Delta'$ and Γ^* has the form $\Gamma', t' : B'$ where $t \doteq_S t'$ and $B \equiv_S B'$. Then there is some permutation ρ such that either (a) $\rho u \doteq_S t$ and $\rho{\cdot}A \equiv_S B' \equiv_S B$, or (b) Γ has the form $\Gamma'', t'' : B''$ where $\rho t'' \doteq_S t$ and $\rho{\cdot}B'' \equiv_S B$. In (a), by Lemma 4.4 there is a derivation

$$\dfrac{\pi_1{}'(n)}{\langle S\rangle\, \Gamma \vdash t : B, t : B, \Delta'}$$

and we conclude by admissible (CR). In (b), we can build a cut-free simple proof of $\langle S\rangle\, \Gamma \vdash \Delta$ of unit size by applying a sequence of (τ) rules to the instance of (Id) $\langle S\rangle\, \Gamma'', \rho t'' : \rho{\cdot}B'' \vdash t : B, \Delta'$. The case in which the instance of (Id/τ) occurs as the left premise of the cut is handled in a similar way, also by Lemma 4.4.

(2) (Case Cut - (S$-$)) We have

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S, S'\rangle\, \Gamma \vdash u : A, \Delta \quad u_1 \doteq_S v_1}}{\langle S\rangle\, \Gamma \vdash u : A, \Delta}\,(\text{S}-) \qquad \dfrac{\pi_2(m)}{\langle S\rangle\, \Gamma, u : A \vdash \Delta}}{\langle S\rangle\, \Gamma \vdash \Delta}$$

We can now build the derivation

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S, S'\rangle\, \Gamma, u : A \vdash \Delta} \qquad \dfrac{\pi_2{}'(m)}{\langle S, S'\rangle\, \Gamma, u : A \vdash \Delta}}{\langle S, S'\rangle\, \Gamma \vdash \Delta \quad u_1 \doteq_S v_1}}{\langle S\rangle\, \Gamma \vdash \Delta}\,(\text{S}-)$$

where $\pi_2'(m)$ is obtained from $\pi_2(m)$ by (W). By induction hypothesis, there is a cut-free derivation of $\langle S, S'\rangle\, \Gamma \vdash \Delta$, so we get the result by (S$-$).

(3) (Case Cut - (И) We handle the case in which the conclusion of (И) is the left premise of the cut, being the right case handled symmetrically. Hence, we have

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S, t \doteq (\boldsymbol{\nu} x)X, x \# N\rangle\,\Gamma \vdash u : A, \Delta}}{\langle S\rangle\,\Gamma \vdash u : A, \Delta}\,(\text{И}) \qquad \dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma, u : A \vdash \Delta}}{\langle S\rangle\,\Gamma \vdash \Delta}$$

We can now build the derivation

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S, t \doteq (\boldsymbol{\nu} x)X, x \# N\rangle\,\Gamma \vdash u : A, \Delta} \qquad \dfrac{{\pi_2}'(m)}{\langle S, t \doteq (\boldsymbol{\nu} x)X, x \# N\rangle\,\Gamma, u : A \vdash \Delta}}{\langle S, t \doteq (\boldsymbol{\nu} x)X, x \# N\rangle\,\Gamma \vdash \Delta}}{\langle S\rangle\,\Gamma \vdash \Delta}\,(\text{И})$$

where $\pi_2'(m)$ is obtained from $\pi_2(m)$ by (W).

By induction hypothesis, there is a cut-free derivation of $\langle S, t \doteq (\boldsymbol{\nu} x)X, x \# N\rangle\,\Gamma \vdash \Delta$, so we get the conclusion by (И).

(4) (Case 4.LR) We consider here the case in which the left premise of the cut rule is the conclusion of a right logical rule that *does not introduce* the cut formula. We consider the general case of a two-premise rule, but the argument can be replicated for single premise rule like (∀R), or (⇒R), which adds an hypothesis (*e.g.*, Γ_1) to the left context. Hence we have,

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma, \Gamma_1 \vdash u : A, \Delta_1} \qquad \dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma, \Gamma_2 \vdash u : A, \Delta_2} \quad C_S}{\langle S\rangle\,\Gamma \vdash u : A, \Delta}\,(-\mathrm{R}) \qquad \dfrac{\pi_3(k)}{\langle S\rangle\,\Gamma, u : A \vdash \Delta}}{\langle S\rangle\,\Gamma \vdash \Delta}$$

where in general the instance of $(-\mathrm{R})$ may also have some assertions C_S as premises. Now, by (W) we can build derivations

$$\dfrac{\pi_3'(k)}{\langle S\rangle\,\Gamma, u : A, \Gamma_1 \vdash \Delta, \Delta_1} \qquad \dfrac{\pi_3''(k)}{\langle S\rangle\,\Gamma, u : A, \Gamma_2 \vdash \Delta, \Delta_2}$$

hence we can construct the derivations

$$\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma, \Gamma_1 \vdash u : A, \Delta, \Delta_1} \qquad \dfrac{\pi_3'(k)}{\langle S\rangle\,\Gamma, u : A, \Gamma_1 \vdash \Delta, \Delta_1}}{\langle S\rangle\,\Gamma, \Gamma_1 \vdash \Delta, \Delta_1}$$

and

$$\dfrac{\dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma, \Gamma_2 \vdash u : A, \Delta, \Delta_2} \qquad \dfrac{\pi_3''(k)}{\langle S\rangle\,\Gamma, u : A, \Gamma_2 \vdash \Delta, \Delta_2}}{\langle S\rangle\,\Gamma, \Gamma_2 \vdash \Delta, \Delta_2}$$

By induction hypothesis, there are cut-free derivations for the sequents $\langle S\rangle\,\Gamma, \Gamma_1 \vdash \Delta, \Delta_1$ and $\langle S\rangle\,\Gamma, \Gamma_2 \vdash \Delta, \Delta_2$. Hence, by $(-\mathrm{R})$ we can build a cut-free derivation of $\langle S\rangle\,\Gamma \vdash \Delta$, since all possibly required assertions C_S still apply.

(Case 4.LL) The left premise of the cut is the conclusion of a left logical rule that (obviously) *does not introduce* the cut formula. Note that in our proof system all left rules have at most one premise, although some require testing certain assertions C (namely (ИL)). Hence we have in general

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S,S'\rangle\,\Gamma'\vdash u:A,\Delta\quad C_S}}{\langle S\rangle\,\Gamma\vdash u:A,\Delta}\,(-\mathrm{L})\qquad \dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma,u:A\vdash\Delta}}{\langle S\rangle\,\Gamma\vdash\Delta}$$

Using (W) on $\pi_1(n)$ and $\pi_2'(n)$ we can build the derivation

$$\dfrac{\dfrac{\pi_1'(n)}{\langle S,S'\rangle\,\Gamma,\Gamma'\vdash u:A,\Delta}\qquad \dfrac{\pi_2'(m)}{\langle S,S'\rangle\,\Gamma,u:A,\Gamma'\vdash\Delta}}{\langle S,S'\rangle\,\Gamma,\Gamma'\vdash\Delta}$$

By induction hypothesis, we obtain a cut-free derivation of $\langle S,S'\rangle\,\Gamma,\Gamma'\vdash\Delta$. Since $C_{S,S'}$ holds, by $(-\mathrm{L})$ we obtain a cut-free derivation of $\langle S\rangle\,\Gamma\vdash\Delta$.

(Case 4.RR) The right premise of the cut is the conclusion of a right logical rule that (obviously) does not introduce the cut formula. This case is identical to (Case 4.LL) above.

(Case 4.RL) The right premise of the cut is the conclusion of a left logical rule that does not introduce the cut formula. This case is identical to (Case 4.LR) above.

(5) We now consider all cases where the premises of the cut are conclusions of (left and right) logical rules, both introducing the cut formula. Then the rule that occurs in the left (resp. left) premise is a right-rule (resp. left-rule). We consider the various possible rule pairs, there is one such pair for each logical connective.

(Case of 0) We have

$$\dfrac{\dfrac{v\doteq_S \mathbf{0}}{\langle S\rangle\,\Gamma\vdash v:\mathbf{0},\Delta}\qquad \dfrac{\dfrac{\pi_1(n)}{\langle S,v\doteq\mathbf{0}\rangle\,\Gamma\vdash\Delta}}{\langle S\rangle\,\Gamma,v:\mathbf{0}\vdash\Delta}}{\langle S\rangle\,\Gamma\vdash\Delta}$$

Since $v\doteq_S\mathbf{0}$, by (CS) we have

$$\dfrac{{\pi_1}'(n)}{\langle S\rangle\,\Gamma\vdash\Delta}$$

(Case of $\mathtt{F}$) We have

$$\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma\vdash v:\mathtt{F},\Delta}\quad \langle S\rangle\,\Gamma,v:\mathtt{F}\vdash\Delta}{\langle S\rangle\,\Gamma\vdash\Delta}$$

Then $\pi_1(n)$ is a cut-free derivation of $\langle S\rangle\,\Gamma\vdash\Delta$.

(Case of $|$) We have

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma\vdash u':A,u:A|B,\Delta}\quad \dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma\vdash u'':B,u:A|B,\Delta}}{\langle S\rangle\,\Gamma\vdash u:A|B,\Delta}\qquad \dfrac{\dfrac{\pi_3(k)}{\langle S,u\doteq\mathcal{X}|\mathcal{Y}\rangle\,\Gamma,\mathcal{X}:A,\mathcal{Y}:B\vdash\Delta}}{\langle S\rangle\,\Gamma,u:A|B\vdash\Delta}}{\langle S\rangle\,\Gamma\vdash\Delta}$$

where $u \doteq_S u'|u''$. By (Inst$\mathcal{I}$) with $\{\mathcal{X}{\leftarrow}u'\}$ and $\{\mathcal{Y}{\leftarrow}u''\}$ on π_3 we get ${\pi'}_3(k)$

$$\dfrac{{\pi'}_3(k)}{\langle S, u \doteq u'|u''\rangle\, \Gamma, u' : A, u'' : B \vdash \Delta}$$

(note that by the side condition on (|L) $\mathcal{X}$ and $\mathcal{Y}$ do not occur in S, Γ, Δ). Since $u \doteq_S u'|u''$, by (CS) we get ${\pi''}_3(k)$

$$\dfrac{{\pi''}_3(k)}{\langle S\rangle\, \Gamma, u' : A, u'' : B \vdash \Delta}$$

We now build

$$\dfrac{\langle S\rangle\, \Gamma \vdash u' : A, u : A|B, \Delta \quad \langle S\rangle\, \Gamma, u : A|B \vdash \Delta}{\langle S\rangle\, \Gamma \vdash u' : A, \Delta}$$

and

$$\dfrac{\langle S\rangle\, \Gamma \vdash u'' : B, u : A|B, \Delta \quad \langle S\rangle\, \Gamma, u : A|B \vdash \Delta}{\langle S\rangle\, \Gamma \vdash u'' : B, \Delta}$$

By induction hypothesis, these cuts can be eliminated. By (W) from the derivation of $\langle S\rangle\, \Gamma \vdash u' : A, \Delta$ above, we obtain a derivation of $\langle S\rangle\, \Gamma, u'' : B \vdash u' : A, \Delta$. Now we construct

$$\dfrac{\begin{matrix}\vdots \\ \langle S\rangle\, \Gamma \vdash u'' : B, \Delta\end{matrix} \qquad \dfrac{\begin{matrix}\vdots \\ \langle S\rangle\, \Gamma, u'' : B \vdash u' : A, \Delta\end{matrix} \quad \dfrac{{\pi''}_3(k)}{\langle S\rangle\, \Gamma, u' : A, u'' : B \vdash \Delta}}{\langle S\rangle\, \Gamma, u'' : B \vdash \Delta}}{\langle S\rangle\, \Gamma \vdash \Delta}$$

By induction hypothesis, these two cuts can be successively eliminated.

(Case of $\wedge$) We have

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\, \Gamma \vdash u : A, \Delta} \quad \dfrac{\pi_2(m)}{\langle S\rangle\, \Gamma \vdash u : B, \Delta}}{\langle S\rangle\, \Gamma \vdash u : A \wedge B, \Delta} \quad \dfrac{\dfrac{\pi_3(k)}{\langle S\rangle\, \Gamma, u : A, u : B \vdash \Delta}}{\langle S\rangle\, \Gamma, u : A \wedge B \vdash \Delta}}{\langle S\rangle\, \Gamma \vdash \Delta}$$

By (W) on $\pi_1(n)$ we get

$$\dfrac{{\pi'}_1(n)}{\langle S\rangle\, \Gamma, u : B \vdash u : A, \Delta}$$

We now build

$$\dfrac{\dfrac{{\pi'}_1(n)}{\langle S\rangle\, \Gamma, u : B \vdash u : A, \Delta} \quad \dfrac{\pi_3(k)}{\langle S\rangle\, \Gamma, u : A, u : B \vdash \Delta}}{\langle S\rangle\, \Gamma, u : B \vdash \Delta}$$

Since this a single-cut derivation of $\langle S\rangle\, \Gamma, u : B \vdash \Delta$, by induction hypothesis there is a cut-free derivation π' of $\langle S\rangle\, \Gamma, u : B \vdash \Delta$. Hence we build the following single-cut derivation

$$\dfrac{\dfrac{\pi_2(m)}{\langle S\rangle\, \Gamma \vdash u : B, \Delta} \quad \dfrac{\pi'}{\langle S\rangle\, \Gamma, u : B \vdash \Delta}}{\langle S\rangle\, \Gamma \vdash \Delta}$$

We conclude by the induction hypothesis.

(Case of ⇒) We have

$$\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma, u:A \vdash u:B,\Delta}}{\langle S\rangle\,\Gamma \vdash u:A\Rightarrow B,\Delta} \quad \dfrac{\dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma \vdash u:A,\Delta} \quad \dfrac{\pi_3(k)}{\langle S\rangle\,\Gamma, u:B \vdash \Delta}}{\langle S\rangle\,\Gamma, u:A\Rightarrow B \vdash \Delta}$$
$$\overline{\qquad\qquad\langle S\rangle\,\Gamma \vdash \Delta\qquad\qquad}$$

By (W) on $\pi_2(m)$ we get

$$\dfrac{{\pi'}_2(m)}{\langle S\rangle\,\Gamma \vdash u:A, u:B,\Delta}$$

We now build

$$\dfrac{\dfrac{\pi_3(k)}{\langle S\rangle\,\Gamma, u:B \vdash \Delta} \quad \dfrac{\dfrac{{\pi'}_2(m)}{\langle S\rangle\,\Gamma \vdash u:A, u:B,\Delta} \quad \dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma, u:A \vdash u:B,\Delta}}{\langle S\rangle\,\Gamma \vdash u:B,\Delta}}{\langle S\rangle\,\Gamma \vdash \Delta}$$

Like in the case of $\wedge$ above, we can use the induction hypothesis to eliminate these two cuts in succession.

(Case of ▷) Let

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma, \mathcal{X}:A \vdash v':B,\Delta}}{\langle S\rangle\,\Gamma \vdash u:A\rhd B,\Delta} \quad \dfrac{\dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma, u:A\rhd B \vdash t:A,\Delta} \quad \dfrac{\pi_3(k)}{\langle S\rangle\,\Gamma, u:A\rhd B, t|u:B \vdash \Delta}}{\langle S\rangle\,\Gamma, u:A\rhd B \vdash \Delta}}{\langle S\rangle\,\Gamma \vdash \Delta}$$

where $v' \doteq_S \mathcal{X}|u$. By (Inst$\mathcal{I}$) with $\{\mathcal{X}\leftarrow t\}$ on $\pi_1(n)$ and Lemma 4.4 we get

$$\dfrac{{\pi'}_1(n)}{\langle S\rangle\,\Gamma, t:A \vdash t|u:B,\Delta}$$

since by Lemma 2.8(1) $v'\{\mathcal{X}\leftarrow t\} \doteq_S t|u$ (note that by the side condition on (▷R) $\mathcal{X}$ does not occur in S, Γ, Δ). We can now build

$$\dfrac{\dfrac{\pi'(n+1)}{\langle S\rangle\,\Gamma \vdash u:A\rhd B, t:A,\Delta} \quad \dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma, u:A\rhd B \vdash t:A,\Delta}}{\langle S\rangle\,\Gamma \vdash t:A,\Delta}$$

where $\pi'(n+1)$ is obtained from the left premise of the original cut by (W). In a similar way we construct

$$\dfrac{\dfrac{\pi''(n+1)}{\langle S\rangle\,\Gamma, t|u:B \vdash u:A\rhd B,\Delta} \quad \dfrac{\pi_3(k)}{\langle S\rangle\,\Gamma, u:A\rhd B, t|u:B \vdash \Delta}}{\langle S\rangle\,\Gamma, t|u:B \vdash \Delta}$$

By induction hypothesis, these two cuts can be eliminated. By (W) on the first subderivation above we get

$$\begin{array}{c}\vdots\\ \langle S\rangle\,\Gamma \vdash t:A, t|u:B,\Delta\end{array}$$

We now build the following derivation

$$\dfrac{\begin{matrix}\vdots\\ \langle S\rangle\,\Gamma, t|u:B \vdash \Delta\end{matrix} \qquad \dfrac{\begin{matrix}\vdots\\ \langle S\rangle\,\Gamma\vdash t:A, t|u:B,\Delta\end{matrix}\quad \dfrac{\pi'_1(n)}{\langle S\rangle\,\Gamma,t:A\vdash t|u:B,\Delta}}{\langle S\rangle\,\Gamma\vdash t|u:B,\Delta}}{\langle S\rangle\,\Gamma\vdash\Delta}$$

To conclude, we use the induction hypothesis to eliminate the cut on B, and then the cut on A, like in the cases for $\wedge$ and $\Rightarrow$ above.

(Case of $\circledR$) We have

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma\vdash t:A, u:p\circledR A,\Delta}}{\langle S\rangle\,\Gamma\vdash u:p\circledR A,\Delta}\quad \dfrac{\dfrac{\pi_2(m)}{\langle S, u\doteq(\boldsymbol{\nu}p)\mathcal{X}\rangle\,\Gamma,\mathcal{X}:A\vdash\Delta}}{\langle S\rangle\,\Gamma,u:p\circledR A\vdash\Delta}}{\langle S\rangle\,\Gamma\vdash\Delta}$$

where $u\doteq_S(\boldsymbol{\nu}p)t$. By (Inst$\mathcal{I}$) with $\{\mathcal{X}\leftarrow t\}$ on $\pi_2(m)$ we get

$$\dfrac{\pi'_2(m)}{\langle S, u\doteq(\boldsymbol{\nu}p)t\rangle\,\Gamma,t:A\vdash\Delta}$$

Since $u\doteq_S(\boldsymbol{\nu}p)t$, by (CS) we get

$$\dfrac{\pi''_2(m)}{\langle S\rangle\,\Gamma,t:A\vdash\Delta}$$

By (Cut) we can build

$$\dfrac{\langle S\rangle\,\Gamma\vdash t:A,u:p\circledR A,\Delta\quad \langle S\rangle\,\Gamma,u:p\circledR A\vdash\Delta}{\langle S\rangle\,\Gamma\vdash t:A,\Delta}$$

By induction hypothesis this cut can be eliminated. Then we construct the derivation

$$\dfrac{\dfrac{}{\langle S\rangle\,\Gamma\vdash t:A,\Delta}\quad \dfrac{\pi''_2(m)}{\langle S\rangle\,\Gamma,t:A\vdash\Delta}}{\langle S\rangle\,\Gamma\vdash\Delta}$$

and conclude by the induction hypothesis.

(Case of $\oslash$) We have

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma\vdash t:A,\Delta}}{\langle S\rangle\,\Gamma\vdash u:A\oslash p,\Delta}\quad \dfrac{\dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma,v:A\vdash\Delta}}{\langle S\rangle\,\Gamma,u:A\oslash p\vdash\Delta}}{\langle S\rangle\,\Gamma\vdash\Delta}$$

where $t\doteq_S(\boldsymbol{\nu}p)u$. Since $t\doteq_S(\boldsymbol{\nu}p)u$, by (ClosL) we get

$$\dfrac{\pi'_2(m)}{\langle S\rangle\,\Gamma,t:A\vdash\Delta}$$

Then we construct the derivation

$$\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma \vdash t : A, \Delta} \quad \dfrac{\pi'_2(m)}{\langle S\rangle\,\Gamma, t : A \vdash \Delta}}{\langle S\rangle\,\Gamma \vdash \Delta}$$

This cut can be eliminated by the induction hypothesis.

(Case of $\Diamond$) We have

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma \vdash t : A, u : \Diamond A, \Delta}}{\langle S\rangle\,\Gamma \vdash u : \Diamond A, \Delta} \quad \dfrac{\dfrac{\pi_2(m)}{\langle S, u \to \mathcal{X}\rangle\,\Gamma, \mathcal{X} : A \vdash \Delta}}{\langle S\rangle\,\Gamma, u : \Diamond A \vdash \Delta}}{\langle S\rangle\,\Gamma \vdash \Delta}$$

where $u \to_S t$. By (Inst$\mathcal{I}$) with $\{\mathcal{X} \leftarrow t\}$ on $\pi_2(m)$ we get

$$\dfrac{\pi'_2(m)}{\langle S, u \to t\rangle\,\Gamma, t : A \vdash \Delta}$$

Since $u \to_S t$, by (CS) we get

$$\dfrac{\pi''_2(m)}{\langle S\rangle\,\Gamma, t : A \vdash \Delta}$$

Using (Cut) we can build

$$\dfrac{\langle S\rangle\,\Gamma \vdash t : A, u : \Diamond A, \Delta \quad \langle S\rangle\,\Gamma, u : \Diamond A \vdash \Delta}{\langle S\rangle\,\Gamma \vdash t : A, \Delta}$$

By induction hypothesis this cut can be eliminated. We now build the single-cut derivation

$$\dfrac{\begin{matrix}\vdots\\ \langle S\rangle\,\Gamma \vdash t : A, \Delta\end{matrix} \quad \dfrac{\pi''_2(m)}{\langle S\rangle\,\Gamma, t : A \vdash \Delta}}{\langle S\rangle\,\Gamma \vdash \Delta}$$

and conclude by the induction hypothesis.

(Case of $\forall$) We have

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\,\Gamma \vdash u : A\{x \leftarrow y\}, \Delta}}{\langle S\rangle\,\Gamma \vdash u : \forall x.A, \Delta} \quad \dfrac{\dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma, u : A\{x \leftarrow p\}, u : \forall x.A \vdash \Delta}}{\langle S\rangle\,\Gamma, u : \forall x.A \vdash \Delta}}{\langle S\rangle\,\Gamma \vdash \Delta}$$

where y does not occur free in u, S, Γ, Δ. By (Inst$\mathcal{N}$) with $\{x \leftarrow p\}$ on $\pi_1(n)$ we get

$$\dfrac{\pi'_1(n)}{\langle S\rangle\,\Gamma \vdash u : A\{x \leftarrow p\}, \Delta}$$

Using (Cut) we can build

$$\dfrac{\begin{matrix}\vdots\\ \langle S\rangle\,\Gamma, u : A\{x \leftarrow p\} \vdash u : \forall x.A, \Delta\end{matrix} \quad \dfrac{\pi_2(m)}{\langle S\rangle\,\Gamma, u : A\{x \leftarrow p\}, u : \forall x.A \vdash \Delta}}{\langle S\rangle\,\Gamma, u : A\{x \leftarrow p\} \vdash \Delta}$$

where the left premise comes from the left premise of the original cut by (W). By induction hypothesis, this cut can be eliminated. We now build the following single-cut derivation

$$\dfrac{\dfrac{\pi'_1(n)}{\langle S\rangle\, \Gamma \vdash u : A\{x{\leftarrow}p\}, \Delta} \quad \begin{array}{c}\vdots\\ \langle S\rangle\, \Gamma, u : A\{x{\leftarrow}p\} \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash \Delta}$$

and conclude by the induction hypothesis.

(Case of И) We have

$$\dfrac{\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\, \Gamma \vdash u : A\{z{\leftarrow}p\}, u : \text{И}z.A, \Delta}}{\langle S\rangle\, \Gamma \vdash u : \text{И}z.A, \Delta} \quad \dfrac{\dfrac{\pi_2(m)}{\langle S\rangle\, \Gamma, u : A\{z{\leftarrow}q\}, u : \text{И}z.A \vdash \Delta}}{\langle S\rangle\, \Gamma, u : \text{И}z.A \vdash \Delta}}{\langle S\rangle\, \Gamma \vdash \Delta}$$

where $p \,\#_S\, \text{И}z.A$ and $q \,\#_S\, \text{И}z.A$, and $u \doteq_S (\boldsymbol{\nu} p)u'$ and $u \doteq_S (\boldsymbol{\nu} q)u''$. Hence we have $p \,\#_S\, \mathit{fpv}(A)$ and $q \,\#_S\, \mathit{fpv}(A)$. We can now build the derivation

$$\dfrac{\dfrac{\pi_1(n)}{\langle S\rangle\, \Gamma \vdash u : A\{z{\leftarrow}p\}, u : \text{И}z.A, \Delta} \quad \dfrac{\pi'(m+1)}{\langle S\rangle\, \Gamma, u : \text{И}z.A \vdash u : A\{z{\leftarrow}p\}, \Delta}}{\langle S\rangle\, \Gamma \vdash u : A\{z{\leftarrow}p\}, \Delta}$$

where the right premise is obtained from the right premise of the initial cut by (W). Symmetrically, we can build the derivation

$$\dfrac{\dfrac{\pi''(n+1)}{\langle S\rangle\, \Gamma, u : A\{z{\leftarrow}q\} \vdash u : \text{И}z.A, \Delta} \quad \dfrac{\pi_2(m)}{\langle S\rangle\, \Gamma, u : A\{z{\leftarrow}q\}, u : \text{И}z.A \vdash \Delta}}{\langle S\rangle\, \Gamma, u : A\{z{\leftarrow}q\} \vdash \Delta}$$

where the left premise is obtained from the left premise of the initial cut by (W). These two cuts can be eliminated by the induction hypothesis. Since $A\{z{\leftarrow}q\} \equiv_S \{p{\leftrightarrow}q\}{\cdot}A\{z{\leftarrow}p\}$ and $(p \leftrightarrow q)u \doteq_S u$ by (Swap Erase), by Lemma 4.4 there is a derivation of

$$\Gamma \vdash u : A\{z{\leftarrow}q\}, \Delta$$

Then, we build the single-cut derivation

$$\dfrac{\begin{array}{c}\vdots\\ \langle S\rangle\, \Gamma \vdash u : A\{z{\leftarrow}q\}, \Delta\end{array} \quad \begin{array}{c}\vdots\\ \langle S\rangle\, \Gamma, u : A\{z{\leftarrow}q\} \vdash \Delta\end{array}}{\langle S\rangle\, \Gamma \vdash \Delta}$$

and conclude by the induction hypothesis. ∎

Theorem 4.5 [Cut Elimination] *If a sequent has a first-order derivation then it has a derivation without instances of the* (Cut) *rule.*

Proof. We prove the result for CF, the result for the basic system then follows by Proposition 7.11, noting that the transformations between the basic system and CF do not introduce any new cuts. If the sequent has a derivation, by Lemma 7.10 it has an atomic derivation, and then a simple derivation π, by Lemma 4.4 (taking ρ to be the identity permutation). Now, by induction on the number of instances of (Cut) we build a cut-free simple derivation of the same sequent by iterating Lemma 7.13 for each minimal single-cut subderivation of the derivation π. ∎