Acyclic Solos and Differential Interaction Nets*

Thomas Ehrhard
Thomas.Ehrhard@pps.jussieu.fr
Preuves Programmes Systèmes
CNRS – Université Paris 7
France

Olivier Laurent
Olivier.Laurent@pps.jussieu.fr
Preuves Programmes Systèmes
CNRS – Université Paris 7
France

October 20, 2008

Abstract

We present a restriction of the solos calculus which is stable under reduction and expressive enough to contain an encoding of the pi-calculus. As a consequence, it is shown that equalizing names that are already equal is not required by the encoding of the pi-calculus. In particular, the induced solo diagrams bear an acyclicity property that induces a faithful encoding into differential interaction nets. This gives a (new) proof that differential interaction nets are expressive enough to contain an encoding of the pi-calculus.

All this is worked out in the case of finitary (replication free) systems without sum, match nor mismatch.

Keywords: solos calculus, pi-calculus, prefix, typing, differential interaction nets.

ACM classification: F.3.2 Semantics of Programming Languages — Process models ; F.4.1 Mathematical Logic — Lambda calculus and related systems; F.1.2 Modes of Computation — Parallelism and concurrency.

The question of extending the Curry-Howard correspondence (between the λ-calculus and intuitionistic logic) to concurrency theory is a long-standing open problem. We developed in a previous paper [EL08] a translation between the π-calculus [MPW92] and differential interaction nets [ER06]. This has shown that differential linear logic — a logical system whose sequent calculus has been obtained by the first author from a precise analysis of some denotational models of linear logic based on vector spaces [Ehr02] — is a reasonable candidate for a Curry-Howard correspondence with concurrent computation, since differential interaction nets appear to be expressive enough to represent key concurrency primitives as represented in the π-calculus.

Let us tell a bit more about the genesis of this translation. We discovered the notion of *communication areas* as particular differential interaction nets able to represent some communication primitives. However, due to the very asynchronous flavour of differential interaction nets, it was not immediate that additional constructions such as prefixing could be easily encoded. This led us to have a look at the solos calculus.

The *solos calculus* [LV03] has allowed to prove how action prefixes (thus sequentiality constraints) can be encoded in a calculus without prefixes. This is done by encoding the π-calculus

*Partially supported by the French ANR project CHOCO.

into the solos calculus. These two calculi differ on the way they handle name passing. As in the *fusion calculus* [PV98], the solos calculus defines communication by unification of names, whereas the π-calculus uses substitution of a bound name with another one.

Even if their behaviours are very similar, there is a mismatch between the solos calculus and communication areas with respect to the identification of a name x with itself during reduction. Note that such an identification never occurs in the π-calculus. In the solos calculus, if x has to be identified with itself, it is just considered as a dummy operation since we already have $x = x$. Communication areas on their side keep track of this identification through an explicit link (which is not erased by reduction) connecting the communication area associated with x to itself.

We then had two possibilities: to follow the intuition coming from the translation of the π-calculus into the solos calculus to define a translation of the π-calculus into differential interaction nets (this is the methodology of [EL08]), or to find a fragment of the solos calculus which is expressive enough to contain the image of the π-calculus but which does not rely on behaviours represented differently in differential interaction nets (this is the goal of the present paper).

From this point of view, the main result of this work is to provide an alternative proof of the expressive power of differential interaction nets by means of the solos calculus. The first half of the paper will be devoted to the introduction of the required calculi (π-calculus, solos calculus and solo diagrams) together with a simple translation of solo diagrams [LPV01] (the graphical syntax for the solos calculus) into differential interaction nets extracted from the material presented in [EL08]. In particular Section 3 is almost copy-pasted from [EL08]. We will finish the first part of the paper by giving a sufficient condition on solo diagrams for the translation into differential interaction nets to be a bisimulation.

The main technical contribution of the paper comes in the second half. Since the property that a name should not be unified with itself in the solos calculus is of course *not* preserved under the reduction of solos, we have to find a more clever property. By introducing a simple typing system on solos and by deriving constraints on solos terms typed in this system, we introduce the *acyclic solos calculus*. We prove this restriction to be well behaved with respect to the reduction of the solos calculus. We show that the translation of a π-term is always an acyclic solos term, showing the expressiveness of the system. Finally we prove that the sufficient condition introduced at the end of the first part of the paper is fulfilled by solo diagrams corresponding to acyclic solos terms showing that we obtain a bisimulation with respect to differential interaction nets.

Conventions. Since our goal is to focus on prefixing and sequentiality, we deal with calculi without replications nor recursive definitions, without match/mismatch and without sums.

We do not want to spend time to deal with arbitrary arities in the calculi we consider. This is why we only consider *monadic* π-terms. There are three reasons for that: it makes the presentation simpler, it does not lead to a loss of expressiveness, and finally the polyadic case has already been considered in [EL08]. As a consequence (see the translation in Section 1.3), we are led to consider a *triadic* solos calculus (all the names are of arity exactly 3) and *triadic* solo diagrams (all the multiedges are of arity exactly 3). The more general case of arbitrary arities could easily be obtained by introducing the appropriate sortings on the various calculi.

1 The π-calculus and the solos calculus

In this section we recall the definition of the π-calculus and of the solos calculus we are going to use. We also recall the translation from π to solos given in [LV03].

1.1 The π-calculus

The terms of the (monadic) (finitary) π-calculus are given by:

$$P \quad ::= \quad 0 \quad | \quad u(x).P \quad | \quad \overline{u}\langle x\rangle.P \quad | \quad (P \mid P) \quad | \quad \nu x.P$$

where both $u(x).P$ and $\nu x.P$ bind x.

The *structural congruence* on π-terms is the least congruence containing α-equivalence and:

$$\begin{aligned} 0 \mid P &\equiv P \\ P \mid Q &\equiv Q \mid P \\ (P \mid Q) \mid R &\equiv P \mid (Q \mid R) \\ \nu x.\nu y.P &\equiv \nu y.\nu x.P \\ \nu x.0 &\equiv 0 \\ (\nu x.P) \mid Q &\equiv \nu x.(P \mid Q) && \text{if } x \notin \mathsf{fn}(Q) \end{aligned}$$

The *reduction semantics* of the π-calculus is given by:

$$\frac{}{\overline{u}\langle x\rangle.P \mid u(y).Q \to P \mid Q\,[x/y]}$$

$$\frac{P \to Q}{P \mid R \to Q \mid R} \qquad \frac{P \to Q}{\nu x.P \to \nu x.Q} \qquad \frac{P \equiv P' \qquad P' \to Q' \qquad Q' \equiv Q}{P \to Q}$$

1.2 The solos calculus

Introduced in [LV03], the goal of the solos calculus is to prove the expressiveness of a calculus without prefix construction.

The terms of the (triadic) solos calculus are given by:

$$P \quad ::= \quad 0 \quad | \quad u\,x_1x_2x_3 \quad | \quad \overline{u}\,x_1x_2x_3 \quad | \quad (P \mid P) \quad | \quad (x)\,P$$

where $(x)\,P$ binds x.

The *structural congruence* is the least congruence containing α-equivalence and:

$$\begin{aligned} 0 \mid P &\equiv P \\ P \mid Q &\equiv Q \mid P \\ (P \mid Q) \mid R &\equiv P \mid (Q \mid R) \\ (x)\,(y)\,P &\equiv (y)\,(x)\,P \\ (x)\,0 &\equiv 0 \\ ((x)\,P) \mid Q &\equiv (x)\,(P \mid Q) && \text{if } x \notin \mathsf{fn}(Q) \end{aligned}$$

This equivalence allows us to present terms in the solos calculus in canonical forms: either the 0 process or a bunch of scope constructions $(x_1)\,(x_2)\ldots$ followed by solos in parallel.

The *reduction semantics* of the solos calculus is given by ($\tilde{z}$ stands for $z_1 \ldots z_n$):

$$\frac{}{(\tilde{z})\,(\overline{u}\,x_1x_2x_3 \mid u\,y_1y_2y_3 \mid P) \to P\sigma}$$

where σ is a most general unifier of $x_1x_2x_3$ and $y_1y_2y_3$, such that exactly the names in $\tilde{z}$ are modified (in particular, in each equivalence class of names induced by unification, at most one name is free).

$$\frac{P \to Q}{P \mid R \to Q \mid R} \qquad \frac{P \to Q}{(x)\,P \to (x)\,Q} \qquad \frac{P \equiv P' \qquad P' \to Q' \qquad Q' \equiv Q}{P \to Q}$$

An alternative (but equivalent) definition of this reduction semantics is given in [LV03] together with additional explanations.

For example, in $(x)\,(y)\,(z)\,(w)\,(\overline{u}\,uxy \mid u\,zww \mid \overline{v}\,zuy)$, the only possible reduction is between $\overline{u}\,uxy$ and $u\,zww$. It induces the identifications $u = z$, $x = w$ and $y = w$, thus two equivalence classes $\{u, z\}$ and $\{x, y, w\}$. In the first one, u is free and thus the only possibility is to map z to u. In the second one, all the elements are bound, we choose one of them: y for example (the other choices would lead to structurally congruent results). We consider the unifier containing $z \mapsto u$, $x \mapsto y$, $w \mapsto y$ and which is the identity on the other names. We obtain the reduct $(y)\,\overline{v}\,uuy$.

1.3 From π-terms to solos

In [LV03], the authors give different translations of the fusion calculus [PV98] into solos. We are going to focus on one of them (the one which does not introduce matching). By pre-composing this translation with the canonical embedding of the π-calculus into the fusion calculus: $u(x).P \mapsto (x)\,u\,x.P$, we obtain the translation of the π-calculus into solos we present here.

A π-term P is translated as $[P]$:

$$C_v := (z)\,v\,zzv$$

$$\begin{aligned}
[0]_v &:= 0 \\
[u(x).P]_v &:= (w)\,(y)\,(\overline{v}\,uwy \mid C_y \mid (x)\,(v')\,(w\,xvv' \mid [P]_{v'})) \\
[\overline{u}\langle x\rangle.P]_v &:= (w)\,(y)\,(\overline{v}\,uwy \mid C_y \mid (v')\,(\overline{w}\,xv'v \mid [P]_{v'})) \\
[P \mid Q]_v &:= [P]_v \mid [Q]_v \\
[\nu x.P]_v &:= (x)\,[P]_v
\end{aligned}$$

$$[P] := (v)\,([P]_v \mid C_v)$$

The π-term $\nu x.(\overline{u}\langle x\rangle.0 \mid x(y).0) \mid u(z).\overline{z}\langle t\rangle.0$ is translated as a solos term which is structurally congruent to $(v)\,((x)\,(P \mid Q) \mid R \mid C_v)$ with:

$$\begin{aligned}
P &= (w)\,(y)\,(\overline{v}\,uwy \mid C_y \mid (v')\,\overline{w}\,xv'v) \\
Q &= (w)\,(y')\,(\overline{v}\,xwy' \mid C_{y'} \mid (y)\,(v')\,w\,yvv') \\
R &= (w)\,(y)\,(\overline{v}\,uwy \mid C_y \mid (z)\,(v')\,(w\,zvv' \mid (w)\,(y)\,(\overline{v'}\,zwy \mid C_y \mid (v'')\,\overline{w}\,tv''v')))
\end{aligned}$$

By applying reduction, we obtain in particular the following reducts (up to structural congruence):

$$(v)\,((x)\,((C_v \mid (v')\,\overline{u}\,xv'v) \mid Q) \mid R)$$
$$(v)\,((x)\,(((v')\,\overline{u}\,xv'v) \mid Q) \mid C_v \mid (z)\,(v')\,(u\,zvv' \mid (w)\,(y)\,(\overline{v'}\,zwy \mid C_y \mid (v'')\,\overline{w}\,tv''v')))$$
$$(v)\,(x)\,(Q \mid C_v \mid ((w)\,(y)\,(\overline{v}\,xwy \mid C_y \mid (v'')\,\overline{w}\,tv''v)))$$
$$(v)\,(x)\,(C_v \mid (y)\,(v')\,x\,yvv' \mid ((w)\,(y)\,(\overline{v}\,xwy \mid C_y \mid (v'')\,\overline{w}\,tv''v)))$$
$$(v)\,(x)\,((y)\,(v')\,x\,yvv' \mid C_v \mid (v'')\,\overline{x}\,tv''v)$$
$$(v)\,C_v$$

2 Solo diagrams

To make the relation with differential interaction nets simpler, we go through the graphical syntax associated with the solos calculus: solo diagrams [LPV01].

A (triadic) *solo diagram* is given by a finite set of nodes and a finite multiset of (ternary) multiedges (directed edges with a list of three nodes as source and one node as target). A node is tagged as either *free* or *bound*. A multiedge is tagged as either *input* or *output*. Any node must be a source or a target of a multiedge.

2.1 Reduction

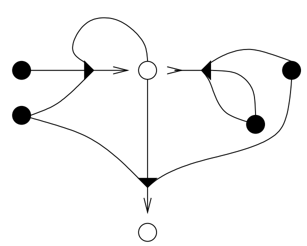
The reduction of solo diagrams is given:

- by choosing two multiedges e_1 and e_2 of opposite polarities (one input and one output) with the same target (we call them *dual multiedges*) and with respective sources $[\mathsf{n}_1, \mathsf{n}_2, \mathsf{n}_3]$ and $[\mathsf{m}_1, \mathsf{m}_2, \mathsf{m}_3]$,
- and by identifying the two nodes n_1 and m_1, the two nodes n_2 and m_2, and the two nodes n_3 and m_3 (with the appropriate constraints on the freeness of nodes: two free nodes are not allowed to be identified).

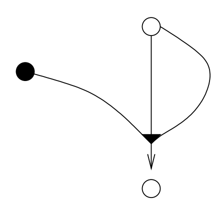
Nodes which are not anymore source or target of a multiedge are removed.

In the graphical representation, we draw free nodes as white dots, bound nodes as black dots, output edges with an outgoing arrow and input edges with an ingoing arrow.

For example:



reduces to



2.2 Relation with solos

A term of the solos calculus can be easily translated into a solo diagram. Free names as translated as free nodes, bound names as bound nodes and solos as multiedges:

- The 0 term is translated as the empty graph with no edge.
- The solo $u\,x_1x_2x_3$ is translated as the graph with free nodes corresponding to elements of $\{u, x_1, x_2, x_3\}$, and with one input multiedge with source $[\mathsf{x}_1, \mathsf{x}_2, \mathsf{x}_3]$ and target u (where x_i is the node corresponding to the name x_i and u is the node corresponding to the name u).
- The solo $\overline{u}\,x_1x_2x_3$ is translated in the same way with an output multiedge.
- The parallel composition $P \mid Q$ is obtained by "graph union": union of the sets of nodes (*i.e.* nodes corresponding to the same name are identified) and sum of the multisets of multiedges.
- The restriction $(x)\,P$ corresponds to turning the free node corresponding to the name x (if any) into a bound node.

As shown in [LPV01], two solos terms are structurally congruent if and only if the corresponding solo diagrams are isomorphic. Moreover reduction in solo diagrams reflects faithfully the reduction of the solos calculus.

The two solo diagrams of Section 2.1 respectively correspond to $(x)\,(y)\,(z)\,(w)\,(\overline{u}\,uxy \mid u\,zww \mid \overline{v}\,zuy)$ and $(y)\,\overline{v}\,uuy$.

2.3 Labeled transition system

We follow the same methodology as in [EL08] by putting labels on multiedges to make possible the distinction between them.

We fix a countable set $\mathcal{L}$ of labels to be used for all our labeled transition systems.

A solo diagram is *labeled* if its multiedges are equipped with different labels belonging to $\mathcal{L}$.

The objects of the labeled transition system $\mathbb{S}_{\mathcal{L}}$ are labeled solo diagrams and transitions are labeled by pairs of distinct elements of $\mathcal{L}$. Let G and H be two labeled solo diagrams, $G \xrightarrow{l\overline{m}} H$ if H is obtained from G by applying a reduction step to the input multiedge labeled l and to the output multiedge labeled m (the labels of the remaining multiedges of H must be the same as for the corresponding ones in G).

2.4 Solo diagrams with identifications

In order to compare solo diagrams with differential interaction nets, it will be useful to decompose the reduction of solo diagrams by introducing the notion of *solo diagrams with identifications*. Solo diagrams with identifications are solo diagrams equipped with a finite set of undirected edges (usual binary edges, not multiedges). These edges are called *identification edges*.

We can decompose the reduction of solo diagrams we have presented above by using identification edges:

R1. choose two dual multiedges e_1 and e_2 (*i.e.* of opposite polarities and with the same target) with respective sources $[\mathsf{n}_1, \mathsf{n}_2, \mathsf{n}_3]$ and $[\mathsf{m}_1, \mathsf{m}_2, \mathsf{m}_3]$;

R2. build the solo diagram with identifications $G[e_1, e_2]$ obtained from G by erasing e_1 and e_2 and by introducing three identification edges: between n_1 and m_1, between n_2 and m_2, and between n_3 and m_3;

R3. contract the graph $G[e_1, e_2]$ by (repeatedly) choosing an identification edge and by identifying the two (or one) nodes it connects if at least one of them is bound.

The reduction succeeds (*i.e.* is a valid reduction of solo diagrams) if we reach a solo diagram with no remaining identification edge.

In the graphical representation, we draw identification edges as dashed edges.

If we refine the reduction of the example of Section 2.1 by means of solo diagrams with identification, the results of step (R2) and then of one application of step (R3) are:

3 Differential Interaction Nets

We first recall the general syntax of interaction nets, as introduced in [Laf95]. See also [ER06] for more details.

3.1 The general formalism of interaction nets

Assume we are given a set of *symbols* and that an arity (a non-negative integer) and a typing rule is associated with each symbol, this typing rule being a list $(A_0, A_1, \ldots, A_n)$ of types (where n is the arity associated with the symbol; types are formulae of some system of linear logic). A *net* is made of *cells*. With each cell γ is associated exactly one symbol and therefore an arity n and a typing rule $(A_0, A_1, \ldots, A_n)$. Such a cell γ has one *principal port* p_0 and n *auxiliary ports* $p_1, \ldots, p_n$. A net has also a finite set of *free ports*. All these ports (the free ports and the ports associated with cells) have to be pairwise distinct and a set of *wires* is given. This wiring is a family of pairwise disjoint sets of ports of cardinality 2 (ordinary wires) or 0 (loops), and the union of these wires must be equal to the set of all ports of the net. An *oriented wire* of the net is an ordered pair (p_1, p_2) where $\{p_1, p_2\}$ is a wire. In a net, a type is associated with each oriented wire, in such a way that if A is associated with (p_1, p_2), then $A^\perp$ is associated with (p_2, p_1). Last, the typing rules of the cells must be respected in the sense that for each cell γ of arity n, whose ports are $p_0, p_1, \ldots, p_n$ and typing rule is $(A_0, A_1, \ldots, A_n)$, denoting by $p'_0, p'_1, \ldots, p'_n$ the ports of the net uniquely defined by the fact that the sets $\{p_i, p'_i\}$ are wires (for $i = 0, 1, \ldots, n$), then the oriented wires (p_0, p'_0), (p'_1, p_1),…,(p'_n, p_n) have types A_0, A_1, …, A_n respectively.

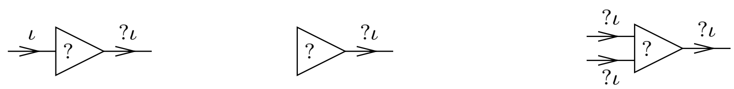
3.2 Presentation of the cells

Our nets will be typed using a type system which corresponds to the untyped lambda-calculus. This system is based on a single type symbol o (the type of outputs), subject to the following recursive equation $o = ?o^\perp \parr o$. We set $\iota = o^\perp$, so that $\iota = !o \otimes \iota$ and $o = ?\iota \parr o$.

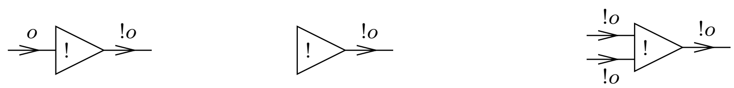
In the present setting, there are ten symbols: par (arity 2), bottom (arity 0), tensor (arity 2), one (arity 0), dereliction (arity 1), weakening (arity 0), contraction (arity 2), codereliction (arity 1), coweakening (arity 0) and cocontraction (arity 2). We present now the various cell symbols, with their typing rules, in a pictorial way. The principal port of a cell is located at one of the angles of the triangle representing the cell, the other ports are located on the opposite edge. We put often a black dot to locate the auxiliary port number 1.

Multiplicative cells. The *par* and *tensor* cells, as well as their "nullary" versions *bottom* and *one* are as follows:

Exponential cells. They are typed according to a strictly polarized discipline. Here are first the *why not* cells, which are called *dereliction*, *weakening* and *contraction*:



and then the *bang* cells, called *codereliction*, *coweakening* and *cocontraction*:



Nets. A *simple net* is a typed interaction net, in the signature we have just defined.

A *net* is a finite formal sum of simple nets having all the same interface. Remember that the interface of a simple net s is the set of its free ports, together with the mapping associating with each free port the type of the oriented wire of s whose ending point is the corresponding port.

Labeled nets. Let $\mathcal{L}_\tau$ be the countable set of labels obtained by adding a distinguished element τ (to be understood as the absence of label) to $\mathcal{L}$. A *labeled simple net* is a simple net where all dereliction and codereliction cells are equipped with labels belonging to $\mathcal{L}_\tau$. Two such labels are either different or equal to τ.

All the nets we consider in this paper are labeled. In our pictures, the labels of dereliction and codereliction cells will be indicated, unless it is τ, in which case the (co)dereliction cell will be drawn without any label.

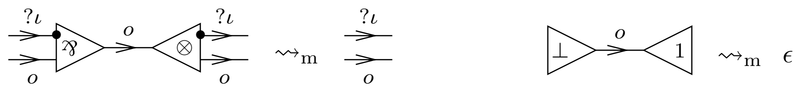
3.3 Reduction rules

We denote by $\mathbf{\Delta}$ the collection of all simple nets, ranged over by the letters s, t, u, with or without subscripts or superscripts, and by $\mathbb{N}\langle\mathbf{\Delta}\rangle$ the collection of all nets (finite sums of simple nets with the same interface), ranged over by the letters S, T, U, with or without subscripts or superscripts. We consider $\mathbf{\Delta}$ as a subset of $\mathbb{N}\langle\mathbf{\Delta}\rangle$ ($s \in \mathbf{\Delta}$ being identified with the sum made of exactly one copy of s).

A *reduction rule* is a subset $\mathcal{R}$ of $\mathbf{\Delta} \times \mathbb{N}\langle\mathbf{\Delta}\rangle$ consisting of pairs (s, S) where s is a simple net made of two cells connected by their principal ports and S is a net that has the same interface as s. This set $\mathcal{R}$ can be finite or infinite. Such a relation is easily extended to arbitrary simple nets ($s \; \mathcal{R} \; T$ if there is $(s_0, u_1 + \cdots + u_n) \in \mathcal{R}$ where s_0 is a subnet of s, each u_i is a simple net and $T = t_1 + \cdots + t_n$ where t_i is the simple net obtained by replacing s_0 by u_i in s). This relation is extended to nets (sums of simple nets): $s_1 + \cdots + s_n$ (where each s_i is simple) is related to T by this extension $\mathcal{R}^\Sigma$ if $T = T_1 + \cdots + T_n$ where, for each i, $s_i \mathrel{\mathcal{R}} T_i$ or $s_i = T_i$. Last, $\mathcal{R}^*$ is the transitive closure of $\mathcal{R}^\Sigma$ (which is reflexive), and we use $\mathcal{R}^+$ for $\mathcal{R}^* \setminus \mathrm{Id}$ (where Id is the identity relation).

3.3.1 Defining the reduction

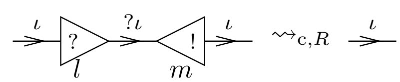
Multiplicative reduction. The first two rules concern the interaction of two multiplicative cells of the same arity.



where ϵ stands for the empty simple net (not to be confused with the net $0 \in \mathbb{N}\langle \mathbf{\Delta} \rangle$, the empty sum, which is not a simple net). The next two rules concern the interaction between a binary and a nullary multiplicative cell.

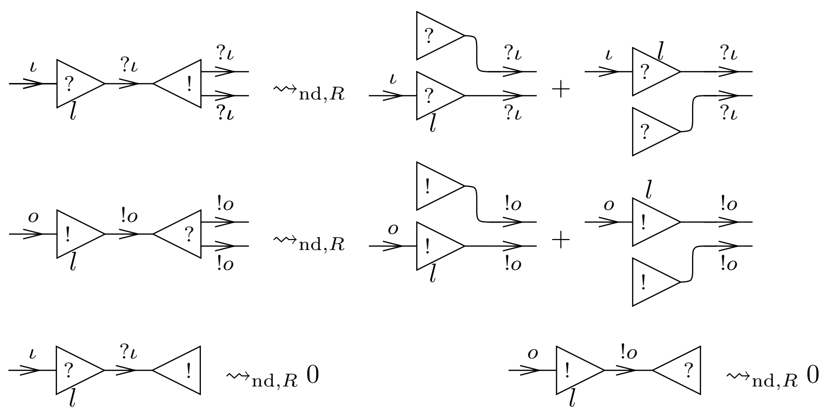
So here the reduction rule (denoted as $\rightsquigarrow_{\mathrm{m}}$) has four elements.

Communication reduction. Let $R \subseteq \mathcal{L}_\tau$. We have the following reductions if $l, m \in R$.

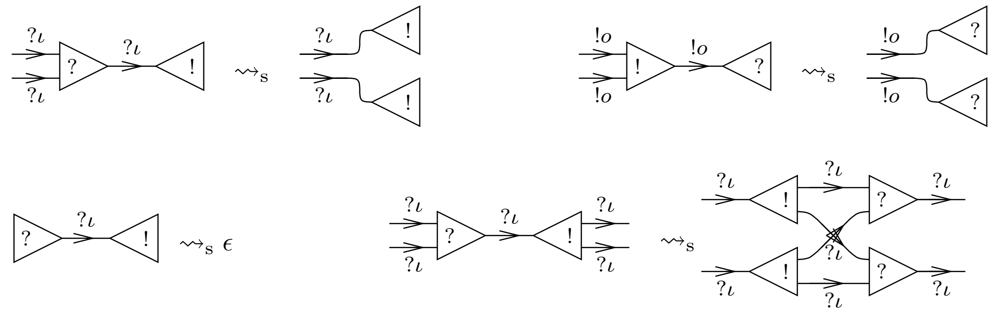


So the set $\rightsquigarrow_{\mathrm{c},R}$ is in bijective correspondence with R^2.

Non-deterministic reduction. Let $R \subseteq \mathcal{L}_\tau$. We have the following reductions if $l \in R$.



Structural reduction.



We use $\sim_{\mathrm{s}}$ for the symmetric and transitive closure of $\rightsquigarrow_{\mathrm{s}}$.

Remark 1. *One can check that we have provided reduction rules for all redexes compatible with our typing system: for any simple net s made of two cells connected through their principal ports, there is a reduction rule whose left member is s. This rule is unique, up to the choice of a set of labels, but this choice has no influence on the right member of the rule.*

3.3.2 Confluence

Theorem 1. *Let $R, R' \subseteq \mathcal{L}_\tau$. Let $\mathcal{R} \subseteq \mathbf{\Delta} \times \mathbb{N}\langle\mathbf{\Delta}\rangle$ be the union of some of the reduction relations $\rightsquigarrow_{\mathrm{c},R}$, $\rightsquigarrow_{\mathrm{nd},R'}$, $\rightsquigarrow_{\mathrm{m}}$ and $\rightsquigarrow_{\mathrm{s}}$. The relation $\mathcal{R}^*$ is confluent on $\mathbb{N}\langle\mathbf{\Delta}\rangle$.*

The proof is essentially trivial since the rewriting relation has no critical pair (see [ER06]). Given $R \subseteq \mathcal{L}_\tau$, we consider in particular the following reduction: $\rightsquigarrow_R = \rightsquigarrow_{\mathrm{m}} \cup \rightsquigarrow_{\mathrm{c},\{\tau\}} \cup \rightsquigarrow_{\mathrm{s}} \cup \rightsquigarrow_{\mathrm{nd},R}$. We set $\rightsquigarrow_{\mathrm{d}} = \rightsquigarrow_\emptyset$ ("d" for "deterministic") and denote by $\sim_{\mathrm{d}}$ the symmetric and transitive closure of this relation.

Some of the reduction rules we have defined depend on a set of labels. This dependence is clearly monotone in the sense that the relation becomes larger when the set of labels increases.

3.3.3 A transition system of simple nets

$\{l, m\}$-neutrality. Let l and m be distinct elements of $\mathcal{L}$. We call (l, m)-*communication redex* a communication redex whose codereliction cell is labeled by l and whose dereliction cell is labeled by m.

We say that a simple net s is $\{l, m\}$-*neutral* if, whenever $s \rightsquigarrow^*_{\{l,m\}} S$, none of the simple summands of S contains an (l, m)-communication redex.

The transition system. We define a labeled transition system $\mathbb{D}_{\mathcal{L}}$ whose objects are simple nets, and transitions are labeled by pairs of distinct elements of $\mathcal{L}$. Let s and t be simple nets, we have $s \xrightarrow{l\overline{m}} t$ if the following holds: $s \rightsquigarrow^*_{\{l,m\}} s_1 + s_2 + \cdots + s_n$ where s_1 is a simple net which contains an (l, m)-communication redex and becomes t when one reduces this redex, and each s_i (for $i > 1$) is $\{l, m\}$-neutral.

Lemma 1. *The relation $\sim_{\mathrm{d}} \subseteq \mathbf{\Delta} \times \mathbf{\Delta}$ is a strong bisimulation on $\mathbb{D}_{\mathcal{L}}$.*

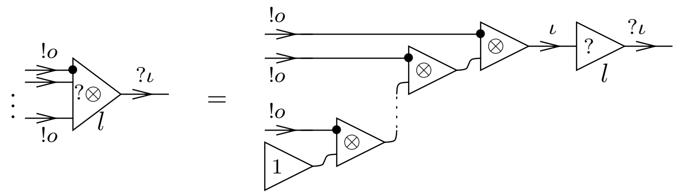


Figure 1: Dereliction-tensor compound cell

Figure 2: Input and output compound cells

3.4 A toolbox for process calculi interpretation

We introduce now a few families of simple nets, which are built using the previously introduced basic cells. They will be used as basic modules for interpreting processes. All of these nets, but the communication areas, can be considered as *compound cells*: in reduction, they behave in the same way as cells of interaction nets.

3.4.1 Compound cells

Generalized contraction and cocontraction. A *generalized contraction cell* or *contraction tree* is a simple net t (with one principal port and a finite number of auxiliary ports) which is either a wire or a weakening cell or a contraction cell whose auxiliary ports are connected to the principal port of other contraction trees, whose auxiliary ports become the auxiliary ports of t. Generalized cocontraction cells (cocontraction trees) are defined dually.

We use the same graphical notations for generalized (co)contraction cells as for ordinary (co)contraction cells, with a "$*$" in superscript to the "!" or "?" symbols to avoid confusions. Observe that there are infinitely many generalized (co)contraction cells of any given arity.

The dereliction-tensor and the codereliction-par cells. Let n be a non-negative integer. We define an n-ary $?\otimes$ compound cell as in Figure 1. It will be decorated by the label of its dereliction cell (if different from τ). The number of tensor cells in this compound cell is equal to n. One defines dually the $!\parr$ compound cell.

The prefix cells. Now we can define the compound cells which will play the main role in the interpretation of solos. Thanks to the above defined cells, all the oriented wires of the nets we shall define will bear type $?\iota$ or $!o$. Therefore, we adopt the following graphical convention: the wires will bear an orientation corresponding to the $?\iota$ type.

The *n-ary input cell* and the *n-ary output cell* are defined in Figure 2, they have $2n$ pairs of auxiliary ports $(\delta_1^+, \delta_1^-, \ldots, \delta_n^+, \delta_n^-)$.

Prefix cells are labeled by the label carried by their outermost $?\otimes$ or $!\parr$ compound cell, if different from τ, the other $?\otimes$ or $!\parr$ compound cells being unlabeled (that is, labeled by τ).

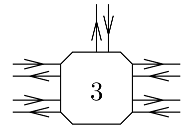


Figure 3: Area of order 3

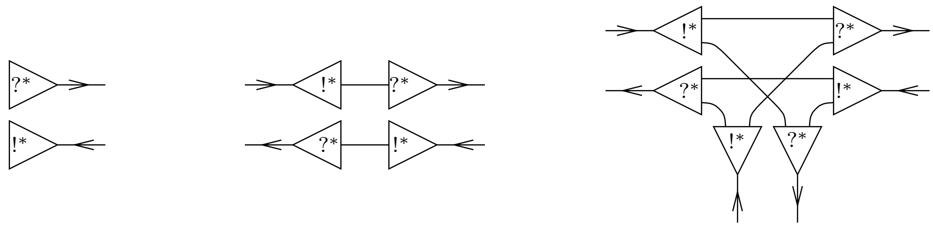


Figure 4: Communication areas of order −1, 0 and 1

3.4.2 Communication areas

Let $n \geq -2$. We define a family of nets with $2(n+2)$ free ports, called *communication areas of order* n, that we shall draw using rectangles with beveled angles. Figure 3 shows how we picture a communication area of order 3.

A communication area of order n is made of $n+2$ pairs of $(n+1)$-ary generalized cocontraction and contraction cells $(\gamma_1^+, \gamma_1^-), \ldots, (\gamma_{n+1}^+, \gamma_{n+1}^-)$, with, for each i and j such that $1 \leq i < j \leq n+2$, a wire from an auxiliary port of γ_i^+ to an auxiliary port of γ_j^- and a wire from an auxiliary port of γ_i^- to an auxiliary port of γ_j^+. We note p_i^+ and p_i^- the principal ports of γ_i^+ and γ_i^-.

So the communication area of order -2 is the empty net ϵ, and communication areas of order -1, 0 and 1 are the structures given in Figure 4.

3.4.3 Useful reductions.

Aggregation of communication areas. One of the nice properties of communication areas is that, when one connects two such areas through a pair of wires, one gets another communication area; if the two areas are of respective orders $p \geq -1$ and $q \geq -1$, the resulting area is of order $p+q$, see Figure 5.

Port forwarding in a net. Let t be a net and p be a free port of t. We say that p *is forwarded in* t if there is a free port q of t such that t is of one of the two shapes given in Figure 6.

Forwarding of derelictions and coderelictions in communication areas. The reduction of Figure 7 shows that derelictions and coderelictions can meet each other, when connected to a common communication area. More precisely, let $l, m \in \mathcal{L}_\tau$, then we have the reduction of Figure 7,

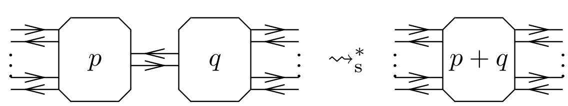


Figure 5: Aggregation, with $p, q \geq -1$

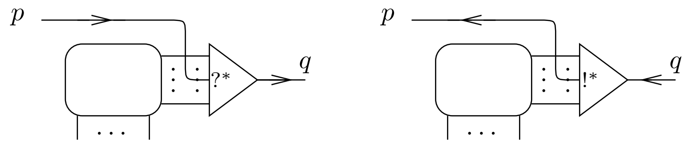


Figure 6: Port forwarding

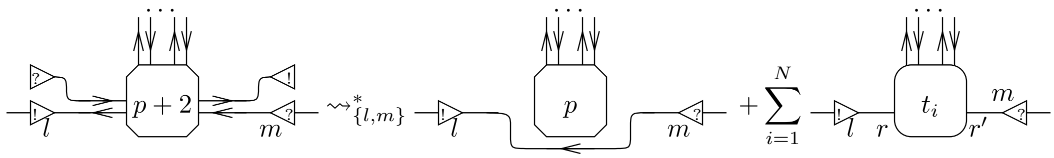


Figure 7: Dereliction and codereliction communicating through a communication area

where N is a non-negative integer (actually, $N = (p+2)^2$) and, in each simple net t_i, both ports r and r' are forwarded.

Reduction of prefixes. Let $l, m \in \mathcal{L}_\tau$. If we connect an n-ary output prefix labeled by m to a p-ary input prefix labeled by l, we obtain a net which reduces by $\rightsquigarrow_{c,\{l,m\}}$ to a net u which reduces by $\rightsquigarrow^*_{\{\tau\}}$ to 0 if $n \neq p$ and to simple wires, in Figure 8, if $n = p$.

4 From solo diagrams to differential interaction nets

4.1 The translation

Relying on the toolbox described above, we define a translation of labeled solo diagrams with identifications into labeled differential interaction nets:

- A node which appears n times as a source of a multiedge, which is k times target of a multiedge and which is p times member of an identification edge is translated as a *communication area* of order $n + k + p - 2$.
- An input multiedge with sources $[\mathsf{x}_1, \mathsf{x}_2, \mathsf{x}_3]$ and target u is translated as a 3-ary *input cell* with principal port u^0 and auxiliary ports $[x_1^0, x_1^1, x_2^0, x_2^1, x_3^0, x_3^1]$. Each pair x_i^0, x_i^1 is connected to a pair of associated ports of the communication area corresponding to x_i. u^0 is connected to a port p^+ of the communication area corresponding to u. A weakening cell is connected to p^-. The label of the prefix cell is the label of the multiedge.

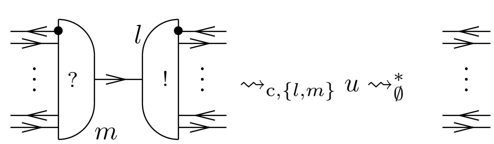


Figure 8: Prefix reduction

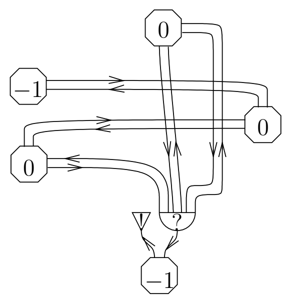
- An output multiedge with sources $[\mathsf{x}_1, \mathsf{x}_2, \mathsf{x}_3]$ and target u is translated as a 3-ary *output cell* with principal port u^0 and auxiliary ports $[x_1^0, x_1^1, x_2^0, x_2^1, x_3^0, x_3^1]$. Each pair x_i^0, x_i^1 is connected to a pair of associated ports of the communication area corresponding to x_i. u^0 is connected to a port p^- of the communication area corresponding to u. A coweakening cell is connected to p^+. The label of the prefix cell is the label of the multiedge.

- An identification edge connecting x_1 and x_2 is translated as a pair of wires connecting a pair (p_1^+, p_1^-) of ports of the communication area corresponding to x_1 with a pair (p_2^-, p_2^+) of ports of the communication area corresponding to x_2.

Since communication areas of order n are not uniquely defined (because generalized contraction and cocontraction cells of a given arity are not unique either), this translation from solo diagrams to differential interaction nets is in fact a relation, which we denote $G \leftrightsquigarrow s$.

Remark 2. *If $G \leftrightsquigarrow s$ then s has no $\rightsquigarrow_{\mathrm{d}}$ redex. Indeed, all the redexes in s are given by a labeled codereliction cell (*i.e. *whose label is not τ) facing a weakening cell, a contraction cell or a labeled dereliction cell (or dually by a dereliction cell facing a coweakening or ...).*

A differential interaction net associated with the second solo diagram with identifications pictured in Section 2.4 is:



4.2 A bisimulation

Our goal is to establish a bisimulation between the labeled transition systems $\mathbb{S}_{\mathcal{L}}$ (Section 2.3) and $\mathbb{D}_{\mathcal{L}}$ (Section 3.3.3). The translation $\leftrightsquigarrow$ does not provide such a bisimulation. We are going to show what the problems are and how to restrict $\mathbb{S}_{\mathcal{L}}$ to get a bisimulation.

4.2.1 Mismatch

The crucial point comes from step (R3) of the reduction of solo diagrams: the contraction of identification edges in solo diagrams corresponds through $\leftrightsquigarrow$ to the aggregation of communication areas as given by the following lemma.

Lemma 2. *If G_1 and G_2 are solo diagrams with identifications, if G_2 is obtained from G_1 by contracting an identification edge (step (R3)) connecting the nodes n_1 and n_2 with $\mathsf{n}_1 \neq \mathsf{n}_2$ and if $G_1 \leftrightsquigarrow s_1$ then $G_2 \leftrightsquigarrow s_2$ where s_2 is obtained from s_1 by aggregating the communication areas corresponding to n_1 and n_2.*

The hypothesis $\mathsf{n}_1 \neq \mathsf{n}_2$ is crucial since two communication areas C_1 and C_2 connected together with a pair of wires reduce to one communication area (see the aggregation of communication areas in Section 3.4.3), *except* if $C_1 = C_2$! We are thus able to encode the reduction of the solos calculus only if we can ensure that we never have to contract an identification edge connecting a node with itself.

A typical example of this problem is given by:

$$(u)\,(x)\,(y)\,(z)\,(y')\,(z')\,(a)\,(b)\,(c)\,(a')\,(b')\,(c')\;\;(\overline{u}\,xyz \mid u\,xy'z' \mid \overline{x}\,abc \mid x\,a'b'c')$$

As shown in Figure 9, while in solo diagrams the loop created after the first step eventually disappears, it remains in differential interaction nets. The consequence being that we have transitions in $\mathbb{S}_{\mathcal{L}}$ while the last differential interaction net has no transitions: the possibility for the two prefix cells to interact "inside the loop", and not only in the direct way, generates summands which are not $\{l, m\}$-neutral (where l and m are the labels of these prefix cells) after reduction.

4.2.2 Restriction

In order to avoid this situation and to be sure we are able to apply Lemma 2, we have to restrict the reduction step (R3) to the case where the edge is not connecting a node with itself. This is equivalent to asking the sub-graph $G_{\mathrm{id}}[e_1, e_2]$ of $G[e_1, e_2]$, containing the identification edges only, to be acyclic. In such a case, we call the reduction step associated with e_1 and e_2 an *acyclic reduction step*.

Another problem comes from the constraint on the freeness of names in the contraction of identification edges in solo diagrams. Our translation into differential interaction nets is forgetting the fact that some nodes are free and others are bound. As a consequence a reduction might happen in the differential interaction net's side without being possible in the solo diagram's side. An example is given by the term $\overline{u}\,xyz \mid u\,x'y'z'$. To avoid this situation we introduce the notion of *acyclic redex*: in a solo diagram, a pair of dual multiedges is an acyclic redex if it is a redex (meaning that the induced freeness conditions are satisfied) and the induced reduction step is an acyclic reduction step.

We define $\mathbb{S}_{\mathcal{L}}^{\mathsf{ac}}$ has the biggest labeled transition system such that:

- objects are objects of $\mathbb{S}_{\mathcal{L}}$ and transitions are transitions of $\mathbb{S}_{\mathcal{L}}$,
- for any object in $\mathbb{S}_{\mathcal{L}}^{\mathsf{ac}}$, all the transitions starting from it in $\mathbb{S}_{\mathcal{L}}$ belong to $\mathbb{S}_{\mathcal{L}}^{\mathsf{ac}}$,
- all pairs of dual multiedges in objects of $\mathbb{S}_{\mathcal{L}}^{\mathsf{ac}}$ are acyclic redexes.

This means that an object G in $\mathbb{S}_{\mathcal{L}}$ belongs to $\mathbb{S}_{\mathcal{L}}^{\mathsf{ac}}$ as soon as none of the paths starting from G in $\mathbb{S}_{\mathcal{L}}$ allows us to reach an object with a non acyclic redex.

A solo diagram belonging to $\mathbb{S}_{\mathcal{L}}^{\mathsf{ac}}$ is called an *acyclic solo diagram*.

Lemma 3 (Diving)**.** *If $G \leftrightsquigarrow s$ and $s \rightsquigarrow^{*}_{\{l,m\}} t{+}T$ with t containing an (l, m)-communication redex, then the codereliction labeled l and the dereliction labeled m are connected to the same communication area C in s and the (l, m)-communication redex is generated by the dereliction/codereliction forwarding through C (see Section 3.4.3).*

Proof. Since s has no $\rightsquigarrow_{\mathrm{d}}$ redex (Remark 2), its only $\rightsquigarrow_{\{l,m\}}$ redexes are involving the codereliction l or the dereliction m.

If s contains an (l, m)-communication redex, we are done (it corresponds to C being the simplest communication area of order 0: two wires).

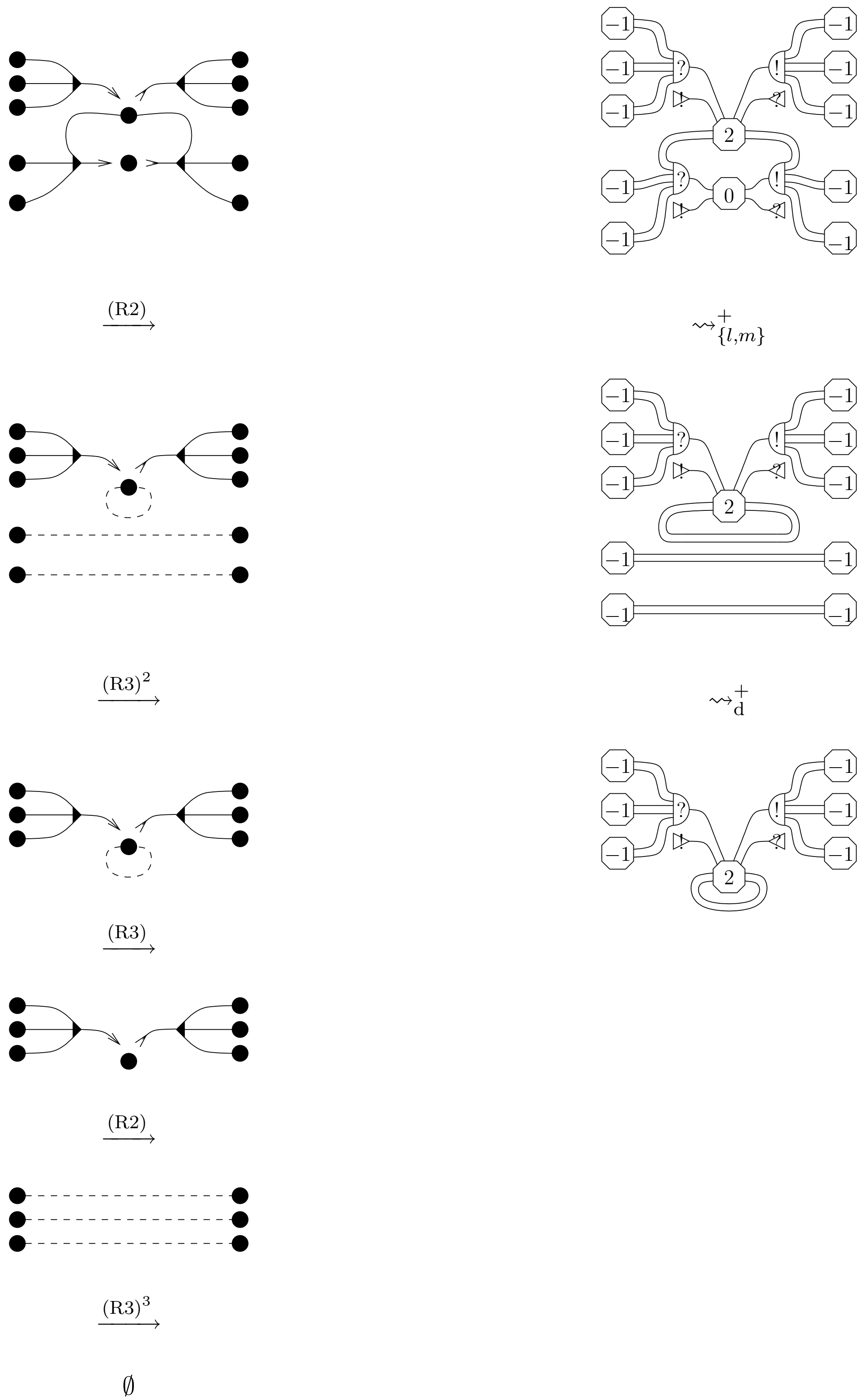


Figure 9: Reductions in solo diagrams and in differential interaction nets

Otherwise, one of them is facing a generalized (co)contraction cell, and the reduction from s to t starts with a "dereliction / generalized cocontraction step" or a "codereliction / generalized contraction step". Let us assume this reduction step involves the codereliction l. After this step, its main port becomes connected to an auxiliary port of a generalized cocontraction cell, or to an auxiliary port of a prefix cell, or to the principal port of a dereliction cell. In all these cases, the obtained net has no $\rightsquigarrow_{\{l\}}$ redex. The same story applies to the dereliction m, which interacts with a generalized cocontraction cell and its main port becomes connected to an auxiliary port of a generalized contraction cell, or to an auxiliary port of a prefix cell, or to the principal port of a codereliction cell. The only way the net t can have an (l, m)-communication redex is if the codereliction l and the dereliction m are facing each other at this point. This means they were connected to the same communication area C in s and what we have done is just forwarding them through C. □

Proposition 1. *If $G \xrightarrow{l\overline{m}} H$ in $\mathbb{S}^{ac}_{\mathcal{L}}$ and $G \leftrightsquigarrow s$, then there are simple nets t and t_0 such that $s \xrightarrow{l\overline{m}} t_0$ in $\mathbb{D}_{\mathcal{L}}$, $H \leftrightsquigarrow t$ and $t_0 \rightsquigarrow_{\mathrm{d}} t$.*

Proof. By hypothesis, there is a communication area in s with two dual prefixes labeled with l and m connected to it. We apply the forwarding of derelictions and coderelictions in communication areas to them. This gives us a sum of simple nets containing a simple net s_1 with an (l, m)-communication redex and other summands where l and m have been forwarded and which are $\{l, m\}$-neutral (Lemma 3).

Let t_0 be the simple net obtained by reducing the (l, m)-communication redex of s_1, and t_1 be the simple net obtained by applying the whole prefix reduction to s_1. It is easy to see that $G[e_1, e_2] \leftrightsquigarrow t_1$. By Lemma 2, each step of acyclic contraction of an identification edge corresponds to the aggregation of the two associated communication areas, thus $H \leftrightsquigarrow t$ with $t_1 \rightsquigarrow^*_{\mathrm{d}} t$ thus $t_0 \rightsquigarrow^*_{\mathrm{d}} t$. □

Proposition 2. *If $s \xrightarrow{l\overline{m}} t_0$ in $\mathbb{D}_{\mathcal{L}}$ and $G \leftrightsquigarrow s$ with $G \in \mathbb{S}^{ac}_{\mathcal{L}}$, then there are a solo diagram H and a simple net t such that $G \xrightarrow{l\overline{m}} H$ in $\mathbb{S}^{ac}_{\mathcal{L}}$, $H \leftrightsquigarrow t$ and $t_0 \rightsquigarrow_{\mathrm{d}} t$.*

Proof. By Lemma 3, G contains two dual multiedges e_1 and e_2 connected to the same node and labeled l and m. t_0 is obtained from a simple net s_1 by reducing an (l, m)-communication redex coming from two prefix cells facing each other. Let t_1 be the simple net obtained by finishing this prefix reduction in t_0, we first show that $G[e_1, e_2] \leftrightsquigarrow t_1$. By Lemma 3 again, the only way the (l, m)-communication redex can have been generated in s_1 is by the forwarding of the codereliction l and the dereliction m through the communication area (of order $p+2$) they are both connected to. This gives a prefix redex and a communication area of order p. By firing this prefix redex, we generate pairs of wires connecting communication areas which exactly correspond to the identification edges of $G[e_1, e_2]$.

Let H be the solo diagram (without identifications) obtained by firing the redex between e_1 and e_2 in G (this is possible since it is an acyclic redex). We iteratively apply Lemma 2 starting from $G[e_1, e_2]$ and t_1 until we reach H. This is possible since $G \in \mathbb{S}^{ac}_{\mathcal{L}}$. Let t be the simple net obtained this way and corresponding to H, we have $t_0 \rightsquigarrow_{\mathrm{d}} t$. □

To get our bisimulation result, we define $\leftrightsquigarrow_d$ as the composition of $\leftrightsquigarrow$ and $\sim_{\mathrm{d}}$: $G \leftrightsquigarrow_d s$ if there exists a simple net s_0 such that $G \leftrightsquigarrow s_0$ and $s_0 \sim_{\mathrm{d}} s$.

Theorem 2 (Bisimulation). *$\leftrightsquigarrow_d$ is a strong bisimulation between $\mathbb{S}^{\mathsf{ac}}_{\mathcal{L}}$ and $\mathbb{D}_{\mathcal{L}}$.*

Proof. If $G \xrightarrow{l\overline{m}} H$ in $\mathbb{S}^{\mathsf{ac}}_{\mathcal{L}}$, $G \leftrightsquigarrow s_0$ and $s_0 \sim_{\mathrm{d}} s$, by Proposition 1 there are simple nets t and t_0 such that $s_0 \xrightarrow{l\overline{m}} t_0$ in $\mathbb{D}_{\mathcal{L}}$, $H \leftrightsquigarrow t$ and $t_0 \rightsquigarrow_{\mathrm{d}} t$. By Lemma 1, there exists a simple net s_1 such that $s \xrightarrow{l\overline{m}} s_1$ and $t_0 \sim_{\mathrm{d}} s_1$. As a consequence $t \sim_{\mathrm{d}} s_1$ thus $H \leftrightsquigarrow_d s_1$.

Conversely, if $s \xrightarrow{l\overline{m}} t_0$ in $\mathbb{D}_{\mathcal{L}}$, $G \leftrightsquigarrow s_0$ and $s_0 \sim_{\mathrm{d}} s$, by Lemma 1 there exists a simple net s_1 such that $s_0 \xrightarrow{l\overline{m}} s_1$ and $t_0 \sim_{\mathrm{d}} s_1$. By Proposition 2, there are a solo diagram H and a simple net t such that $G \xrightarrow{l\overline{m}} H$ in $\mathbb{S}^{\mathsf{ac}}_{\mathcal{L}}$, $H \leftrightsquigarrow t$ and $s_1 \rightsquigarrow_{\mathrm{d}} t$. As a consequence $t \sim_{\mathrm{d}} t_0$ thus $H \leftrightsquigarrow_d t_0$. ☐

The rest of the paper will be devoted to the definition of a sub-calculus of the solos calculus ensuring the acyclicity of the reduction and thus whose translation into solo diagrams lives inside $\mathbb{S}^{\mathsf{ac}}_{\mathcal{L}}$.

5 The acyclic solos calculus

By analysing the translation of the π-calculus into the solos calculus, we define a sub-system of the solos calculus called the *acyclic solos calculus*. The key properties of this calculus are:

- *expressiveness*: it contains the image of the π-calculus through the translation into solos.
- *stability*: it is well-defined with respect to the reduction of solos since it is stable under reduction.
- *acyclicity*: the solo diagram associated with a term of the acyclic solos calculus is an acyclic solo diagram (and thus the bisimulation with differential interaction nets holds!).

The definition of the acyclic solos calculus is given by means of a typing system assigning Emission/Reception polarities to occurrences of names. This typing system allows us to restrict terms to a forest-like structure with reduction occurring only between roots. This is an abstraction of the forest structure induced by a π-term on its translation into solos.

5.1 Types

We consider a system with only two types V and W. We use U for denoting either V or W.

A *typed term* is a term with scopes decorated with types: $(x^U)\,P$.

A typing judgment is of the shape $\Gamma \vdash P$ where Γ contains typing declarations associating types with names and P is a typed term. The typing rules are:

$$\frac{}{\Gamma \vdash 0} \qquad \frac{\Gamma \vdash P \qquad \Gamma \vdash Q}{\Gamma \vdash P \mid Q} \qquad \frac{\Gamma, x : U \vdash P}{\Gamma \vdash (x^U)\,P}$$

$$\frac{}{\Gamma, x : V, y : V, z : W, z' : W \vdash \hat{x}\, zz'y}$$

$$\frac{}{\Gamma, x : W, z : W, y : V, y' : V \vdash \hat{x}\, zyy'}$$

where $\hat{x}$ is either x or $\overline{x}$.

An intuitive way of understanding the types V and W is given by these two mutually recursive definitions:

$$V ::= WWV$$
$$W ::= WVV$$

(more generally, starting from other mutually recursive definitions of a finite set of types, one could derive other type systems of the same kind which would also satisfy the subject reduction property for example).

The main purpose of this typing system is to be able to associate a *communication protocol* with each object occurrence. Communication protocols are E (emission) and R (reception). If we add communication protocols as a decoration on the definitions of V and W:

$$V ::= W^{\mathtt{R}}W^{\mathtt{E}}V^{\mathtt{E}}$$
$$W ::= W^{\mathtt{R}}V^{\mathtt{E}}V^{\mathtt{R}}$$

then any object occurrence of a typed term in context can be decorated with a communication protocol: in a receipting solo whose subject has type U, the decoration is given according to those of the components of the type, in an emitting solo it is given in a dual way: $\mathtt{E} \mapsto \mathtt{R}$ and $\mathtt{R} \mapsto \mathtt{E}$. Examples are given by the image of the translation of the π-calculus in Section 5.1.2.

The intuition behind E and R comes from the difference between the name-passing mechanism in the π-calculus and in the solos calculus. Name-passing by substitution (as in π) entails that a flow of information is given by interaction from an emission prefix to a reception prefix (reception names are substituted with emission names). Name-passing by unification (as in fusion or solos) establishes a perfect symmetry between the agents trying to communicate. Communication protocols E and R are used to trace the flow of information coming from π after translation into solos. Even if communication is symmetric (inside the solos calculus), it is reasonable for a term decorated with communication protocols to understand the unification of $x^{\mathtt{E}}$ with $y^{\mathtt{R}}$ as y being substituted with x.

5.1.1 Preservation by reduction

We first check elementary properties of the typing system with respect to reduction.

Lemma 4 (Free names)**.** *If $\Gamma \vdash P$ then all the free names of P appear in Γ.*

Lemma 5 (Weakening)**.** *If $x \notin \Gamma$ and $x \notin \mathsf{fn}(P)$, $\Gamma \vdash P$ if and only if $\Gamma, x : U \vdash P$.*

Lemma 6 (Substitution)**.** *If $\Gamma, x : U, y : U \vdash P$ then $\Gamma, x : U \vdash P\,[x/y]$.*

Proof. These three lemmas are proved by simple inductions on the typing derivations. ☐

Proposition 3 (Subject reduction)**.**

1. *If $P \equiv Q$ then $\Gamma \vdash P \;\Leftrightarrow\; \Gamma \vdash Q$.*
2. *If P reduces to Q and $\Gamma \vdash P$ then $\Gamma \vdash Q$.*

Proof. 1. We simply have to consider each case of the definition of $\equiv$. The only interesting one is $((x)\,P) \mid Q \equiv (x)\,(P \mid Q)$ with $x \notin \mathsf{fn}(Q)$. We assume $\Gamma \vdash ((x^U)\,P) \mid Q$, this entails $\Gamma \vdash (x^U)\,P$ and $\Gamma \vdash Q$ and thus $\Gamma, x : U \vdash P$. By Lemma 5, we have $\Gamma, x : U \vdash Q$ thus $\Gamma, x : U \vdash P \mid Q$ and finally $\Gamma \vdash (x^U)\,(P \mid Q)$. Conversely, if $\Gamma \vdash (x^U)\,(P \mid Q)$ then $\Gamma, x : U \vdash P \mid Q$, $\Gamma, x : U \vdash P$, $\Gamma, x : U \vdash Q$ thus, by Lemma 5 with $x \notin \mathsf{fn}(Q)$, $\Gamma \vdash Q$ and:

$$\dfrac{\dfrac{\Gamma, x : U \vdash P}{\Gamma \vdash (x^U)\,P} \qquad \Gamma \vdash Q}{\Gamma \vdash ((x^U)\,P) \mid Q}$$

2. The only interesting case is $(\tilde{z})\,(\overline{u}\,x_1x_2x_3 \mid u\,y_1y_2y_3 \mid P) \to P\sigma$. From the hypothesis, we deduce $\Gamma, \tilde{z} : \tilde{U} \vdash \overline{u}\,x_1x_2x_3$ and $\Gamma, \tilde{z} : \tilde{U} \vdash u\,y_1y_2y_3$ and $\Gamma, \tilde{z} : \tilde{U} \vdash P$. By Lemma 6, we have $\Gamma \vdash P\sigma$.

□

5.1.2 Typability of the translation

We prove the typability of the translation of any π-term (see Section 1.3 for the definition of the translation).

If P is a π-term with free names in the set $\mathcal{X} = \{x_1, \ldots, x_k\}$, we define $\Gamma_{\mathcal{X}}$ to be the environment $x_1 : W, \ldots, x_k : W$ and $\Gamma_{\mathcal{X},v}$ to be the environment $\Gamma_{\mathcal{X}}, v : V$. We first show that $\Gamma_{\mathcal{X},v} \vdash [P]_v$.

The process C_v is typable:

$$\dfrac{\overline{\Gamma_{\mathcal{X},v}, z : W \vdash v\,zzv}}{\Gamma_{\mathcal{X},v} \vdash (z^W)\,v\,zzv}$$

The translations of prefixes are typable (see Figure 10).

The cases of $P \mid Q$ and $\nu x.P$ are immediate.

Finally we get the typability of $[P]$ by $\Gamma_{\mathcal{X}} \vdash [P]$:

$$\dfrac{\dfrac{\Gamma_{\mathcal{X},v} \vdash [P]_v \qquad \dfrac{\overline{\Gamma_{\mathcal{X},v}, z : W \vdash v\,zzv}}{\Gamma_{\mathcal{X},v} \vdash (z^W)\,v\,zzv}}{\Gamma_{\mathcal{X},v} \vdash [P]_v \mid C_v}}{\Gamma_{\mathcal{X}} \vdash (v^V)\,([P]_v \mid C_v)}$$

This induces the following protocol decorations on the translations of π-terms:

$$\begin{aligned}
C_v &= (z^W)\,v\,z^{\mathtt{R}}z^{\mathtt{E}}v^{\mathtt{E}} \\
[u(x).P]_v &= (w^W)\,(y^V)\,(\overline{v}\,u^{\mathtt{E}}w^{\mathtt{R}}y^{\mathtt{R}} \mid C_y \mid (x^W)\,(v'^V)\,(w\,x^{\mathtt{R}}v^{\mathtt{E}}v'^{\mathtt{R}} \mid [P]_{v'})) \\
[\overline{u}\langle x\rangle.P]_v &= (w^W)\,(y^V)\,(\overline{v}\,u^{\mathtt{E}}w^{\mathtt{R}}y^{\mathtt{R}} \mid C_y \mid (v'^V)\,(\overline{w}\,x^{\mathtt{E}}v'^{\mathtt{R}}v^{\mathtt{E}} \mid [P]_{v'})) \\
[P \mid Q]_v &= [P]_v \mid [Q]_v \\
[\nu x.P]_v &= (x^W)\,[P]_v \\
[P] &= (v^V)\,([P]_v \mid C_v)
\end{aligned}$$

$$
\dfrac{
  \dfrac{
    \dfrac{}{\Gamma_{\mathcal{X},v}, y : V, w : W \vdash \overline{v}\, uwy}
    \qquad
    \Gamma_{\mathcal{X},v}, y : V, w : W \vdash C_y
  }{\Gamma_{\mathcal{X},v}, y : V, w : W \vdash \overline{v}\, uwy \mid C_y}
  \qquad
  \dfrac{
    \dfrac{
      \dfrac{
        \dfrac{}{\Gamma_{\mathcal{X},v}, y : V, w : W, v' : V, x : W \vdash w\, xvv'}
        \qquad
        \Gamma_{\mathcal{X},v}, y : V, w : W, v' : V, x : W \vdash [P]_{v'}
      }{\Gamma_{\mathcal{X},v}, y : V, w : W, v' : V, x : W \vdash w\, xvv' \mid [P]_{v'}}
    }{\Gamma_{\mathcal{X},v}, y : V, w : W, x : W \vdash (v'^{V})\,(w\, xvv' \mid [P]_{v'})}
  }{\Gamma_{\mathcal{X},v}, y : V, w : W \vdash (x^{W})\,(v'^{V})\,(w\, xvv' \mid [P]_{v'})}
}{
  \dfrac{
    \dfrac{\Gamma_{\mathcal{X},v}, y : V, w : W \vdash \overline{v}\, uwy \mid C_y \mid (x^{W})\,(v'^{V})\,(w\, xvv' \mid [P]_{v'})}{\Gamma_{\mathcal{X},v}, w : W \vdash (y^{V})\,\overline{v}\, uwy \mid C_y \mid (x^{W})\,(v'^{V})\,(w\, xvv' \mid [P]_{v'})}
  }{\Gamma_{\mathcal{X},v} \vdash (w^{W})\,(y^{V})\,\overline{v}\, uwy \mid C_y \mid (x^{W})\,(v'^{V})\,(w\, xvv' \mid [P]_{v'})}
}
$$

$$
\dfrac{
  \dfrac{
    \dfrac{}{\Gamma_{\mathcal{X},v}, y : V, w : W \vdash \overline{v}\, uwy}
    \qquad
    \Gamma_{\mathcal{X},v}, y : V, w : W \vdash C_y
  }{\Gamma_{\mathcal{X},v}, y : V, w : W \vdash \overline{v}\, uwy \mid C_y}
  \qquad
  \dfrac{
    \dfrac{
      \dfrac{}{\Gamma_{\mathcal{X},v}, y : V, w : W, v' : V \vdash \overline{w}\, xv'v}
      \qquad
      \Gamma_{\mathcal{X},v}, y : V, w : W, v' : V \vdash [P]_{v'}
    }{\Gamma_{\mathcal{X},v}, y : V, w : W, v' : V \vdash \overline{w}\, xv'v \mid [P]_{v'}}
  }{\Gamma_{\mathcal{X},v}, y : V, w : W \vdash (v'^{V})\,(\overline{w}\, xv'v \mid [P]_{v'})}
}{
  \dfrac{
    \dfrac{\Gamma_{\mathcal{X},v}, y : V, w : W \vdash \overline{v}\, uwy \mid C_y \mid (v'^{V})\,(\overline{w}\, xv'v \mid [P]_{v'})}{\Gamma_{\mathcal{X},v}, w : W \vdash (y^{V})\,\overline{v}\, uwy \mid C_y \mid (v'^{V})\,(\overline{w}\, xv'v \mid [P]_{v'})}
  }{\Gamma_{\mathcal{X},v} \vdash (w^{W})\,(y^{V})\,\overline{v}\, uwy \mid C_y \mid (v'^{V})\,(\overline{w}\, xv'v \mid [P]_{v'})}
}
$$

Figure 10: Typing derivations for the translations of prefixes.

5.2 Acyclicity

Relying on the decoration with communication protocols induced by the typing system, we are now able to define our restriction of the solos calculus.

Given a typed term P, we first define some properties and relations between names and solos occurring in P. If s is a solo, we write $subj(s)$ for its subject.

$x \in s$	$:=$	*x has an object occurrence in s*
$x^{\mathtt{X}} \in s$	$:=$	*x has an object occurrence in s with communication protocol* $\mathtt{X}$
$s \lhd x$	$:=$	$x^{\mathtt{R}} \in s$
$s \lhd t$	$:=$	$s \lhd subj(t)$
$s \perp t$	$:=$	$subj(s) = subj(t)$*, one is emitting and the other is receipting*
s is a root	$:=$	*there is no t such that $t \lhd s$*

We denote by $\lhd^\star$ the reflexive transitive closure of $\lhd$.

Acyclic terms are typed terms which satisfy the following five properties:

AC1. each name has at most one $\mathtt{R}$-occurrence

AC2. $s \perp t$ implies that s and t are roots

AC3. $s \lhd x$ and $x \in t$ implies $s \lhd^\star t$

AC4. $x^{\mathtt{E}} \in s$ and $s \lhd x$ implies s is a reception

AC5. R-occurrences are bound

The *acyclic solos calculus* is given by restricting terms to acyclic terms.

We make a few remarks and state immediate consequences of this definition:

1. Let us consider the following definition of communication protocols for occurrences of names in a π-term: if the occurrence is an object occurrence in a reception prefix, the communication protocol is $\mathtt{R}$, otherwise it is $\mathtt{E}$. Assuming that the names appearing in a π-term are made as different as possible by α-conversion, we can remark that each name has at most one $\mathtt{R}$-occurrence.

2. (AC2) entails that reduction only occurs between roots.

3. In the case where $\lhd$ induces a forest ordering, (AC3) means that all the $\mathtt{E}$-occurrences of x occur above (with respect to this forest ordering) its $\mathtt{R}$-occurrence (if any).

4. (AC4) says that a name can have both an $\mathtt{E}$-occurrence and an $\mathtt{R}$-occurrence in a solo s only if s is a reception. This is a place where we are breaking the symmetry of the solos calculus between emitting solos and receipting solos. This allows us to rule out processes like $x\, y^{\mathtt{E}} y^{\mathtt{R}} u^{\mathtt{R}} \mid \overline{x}\, z^{\mathtt{R}} z^{\mathtt{E}} v^{\mathtt{E}}$ that would lead to twice the identification of y with z which is almost the same as identifying y with y.

5. (AC5) ensures (with (AC1)) that identification edges are always contractible (no freeness problem).

5.2.1 Preservation by reduction

We consider a reduction from P to Q by an interaction between the solos s_0 and t_0 of P. If s is a solo of P, which is different from s_0 and t_0, we will use the name s also for its unique residue in Q (conversely any solo of Q is the residue of a solo of P). The following three lemmas hold assuming only conditions (AC1), (AC2) and (AC3) on P.

Lemma 7. *No* $\mathtt{R}$*-occurrence of* Q *has been substituted.*

Proof. If a name x involved in the reduction (*i.e.* taking part in the unification, *i.e.* occurring in s_0 or t_0) has its $\mathtt{R}$-occurrence in a solo t of P different from s_0 and t_0 then $t \lhd_P x$ (where $\lhd_P$ denotes the $\lhd$ relation of the process P) and, by (AC3), either $t \lhd_P^+ s_0$ or $t \lhd_P^+ t_0$ thus one of them is not a root of P contradicting (AC2). □

Lemma 8. *If the subject of the solo* s *is modified during this reduction, we have both:*

- *s is a root in Q*
- *s is a root in P or $s_0 \lhd_P s$ or $t_0 \lhd_P s$*

Proof. Let x be the subject of s in P and y be its subject in Q, each of x and y appears in either s_0 or t_0. If s is not a root in Q then there is some t such that $t \lhd_Q s$ thus $t \lhd_P y$ (by Lemma 7). If $y \in s_0$ (and symmetrically if $y \in t_0$) then $t \lhd_P^\star s_0$ (by (AC3)) with $t \neq s_0$ since t belongs to Q thus s_0 is not a root in P contradicting (AC2). So that s is a root in Q.

If s is not a root in P then there is some t such that $t \lhd_P s$. Since the subject of s is modified, it means that an $\mathtt{R}$-occurrence in t is also substituted thus $t \notin Q$ by Lemma 7 and $t = s_0$ or $t = t_0$. □

Lemma 9. *If the reduction introduces an object occurrence of the name* x *in* Q *then* x *has no* $\mathtt{R}$*-occurrence in* Q*.*

Proof. For the reduction to put some x in some t, x must occur in s_0 or t_0. Assume that x occurs in s_0 and that x has an $\mathtt{R}$-occurrence in s in Q (so that $s \lhd_Q x$). By Lemma 7, $s \lhd_P x$ and by (AC3), we have $s \lhd_P^\star s_0$ but, since $s \neq s_0$ (because $s_0 \notin Q$), this would entail that s_0 is not a root in P contradicting (AC2). □

We now turn to the preservation result.

Proposition 4 (Acyclicity preservation)**.** *If* P *is an acyclic solos term and* P *reduces to* Q *then* Q *is an acyclic solos term.*

Proof. We first prove the preservation of conditions (AC1), (AC2) and (AC3), independently of conditions (AC4) and (AC5).

Preservation of condition (AC1) is given by Lemma 7.

We move to condition (AC2). If $s \perp_Q s'$, note that if one of them is not a root then none of them is since $t \lhd s \Rightarrow t \lhd subj(s) \Rightarrow t \lhd subj(s') \Rightarrow t \lhd s'$. Assume that $t \lhd_Q s$ and $t \lhd_Q s'$, if the subject of neither s nor s' has been modified during reduction, then $t \lhd_P s$ and $t \lhd_P s'$ (since the $\mathtt{R}$-occurrence in t of the subject of s and s' has not been modified, by Lemma 7) and P contradicts condition (AC2). If one of the subjects of s or s' has been modified then it is a root in Q by Lemma 8.

Concerning condition (AC3), we assume $s \lhd_Q x$ and $x \in_Q t$. By Lemma 7, $s \lhd_P x$ and by Lemma 9, $x \in_P t$. So that, by (AC3), $s \lhd_P^\star t$. If $s \not\lhd_Q^\star t$, then the path from s to t for the relation

$\lhd_P$ has been broken during reduction, this entails that a solo $t' \neq s$ in this path has got his subject modified. By Lemma 8, $s_0 \lhd_P t'$ or $t_0 \lhd_P t'$ but this is impossible because, since s_0 and t_0 are roots in P and are not in Q, $s \neq s_0$ and $s \neq t_0$.

We now assume that (AC1), (AC2) and (AC3) hold. Condition (AC4) is preserved: if, in Q, an emitting solo s contains both an $\mathtt{R}$-occurrence and an $\mathtt{E}$-occurrence of a name x then, by Lemmas 7 and 9, it is also the case in P. Condition (AC5) is preserved by Lemma 7. $\square$

All this shows that our acyclic solos calculus is well behaved with respect to the reduction of the solos calculus.

5.2.2 Acyclicity of the translation

In order to prove the expressiveness of the acyclic solos calculus, we check that it contains the image of the translation of the π-calculus.

The $\lhd$ relation is the following for translated π-terms:

$$\begin{aligned}
C_v &= (z^W)\, v\, z^{\mathtt{R}} z^{\mathtt{E}} v^{\mathtt{E}} \\
[0]_v &= 0 \\
[u(x).P]_v &= (w^W)\,(y^V)\,(\overline{v}\, u^{\mathtt{E}} w^{\mathtt{R}} y^{\mathtt{R}} \mid C_y \mid (x^W)\,(v'^V)\,(w\, x^{\mathtt{R}} v^{\mathtt{E}} v'^{\mathtt{R}} \mid [P]_{v'})) \\
[\overline{u}\langle x\rangle.P]_v &= (w^W)\,(y^V)\,(\overline{v}\, u^{\mathtt{E}} w^{\mathtt{R}} y^{\mathtt{R}} \mid C_y \mid (v'^V)\,(\overline{w}\, x^{\mathtt{E}} v'^{\mathtt{R}} v^{\mathtt{E}} \mid [P]_{v'})) \\
[P \mid Q]_v &= [P]_v \mid [Q]_v \\
[\nu x.P]_v &= (x^W)\,[P]_v \\
[P] &= (v^V)\,([P]_v \mid C_v)
\end{aligned}$$

where an arrow $s \curvearrowleft t$ represents the relation $s \lhd t$.

Let us first make a few remarks:

- The only subject with no $\mathtt{R}$-occurrence (thus the only subject of a root) in $[P]_v$ is v.
- The free names of P have no $\mathtt{R}$-occurrence in $[P]_v$.
- The free names of $[P]_v$ are v and the free names of P.
- $\lhd$ is a forest on $[P]_v$ (which is one of the reasons for the name "acyclic solos").

From these points, conditions (AC1), (AC2), (AC3), (AC4) and (AC5) are easily verified for $[P]_v$ and thus for $[P]$.

5.3 Acyclic solo diagrams

The last property we have to show is that the solo diagram associated with a term of the acyclic solo calculus is an acyclic solo diagram (see Section 4.2.2).

Let G be such a solo diagram associated with the term P of the acyclic solo calculus, we have to show that any pair of dual edges in G is an acyclic redex. This is enough to show that G belongs to $\mathbb{S}^{\mathtt{ac}}_{\mathcal{L}}$. Indeed, if G reduces to H, there exists a term Q which translates into H and such that P reduces to Q. It follows that H is acyclic by preservation of acyclicity through reduction in the solos calculus.

We consider two dual multiedges e_1 and e_2 in G. We want to show that the graph $G_{\text{id}}[e_1, e_2]$ (sub-graph of $G[e_1, e_2]$ with identification edges only, as introduced in Section 4.2.2) is acyclic and does not generate any freeness problem in the contraction of identification edges.

By construction, edges in $G_{\text{id}}[e_1, e_2]$ are connecting two nodes which correspond to occurrences of names in the term P which are going to be unified. By definition of the acyclic solos calculus, one of these occurrences is an E-occurrence and the other one is a R-occurrence. We consider the directed graph G' obtained by orienting the edges of $G_{\text{id}}[e_1, e_2]$ towards R-occurrences.

We first prove the following lemma about directed and non-directed graphs.

Lemma 10. *Let g be a finite directed graph and g_s be the underlying non-directed graph. If g_s is connected, if g has a root (that is a node without incoming edge) and if each node of g has at most one incoming edge then g_s is acyclic.*

Proof. By induction on the number of nodes of g. We consider the connected components of the graph obtained by removing the root r of g and the corresponding edges. We apply the induction hypothesis to each of them: we just have to check that they all have a root which is the case since all of them contain a node which had an incoming edge coming from r and thus is now a root. This entails that all these graphs are acyclic and by adding back r (with at most one edge towards each root) we get an acyclic graph. □

We prove that the connected components of $G_{\text{id}}[e_1, e_2]$ equipped with the orientation of G' satisfy the hypotheses of Lemma 10 and thus that $G_{\text{id}}[e_1, e_2]$ is acyclic. Let us assume e_1 is the input multiedge in the redex between e_1 and e_2 in G (since solos in a solos term are in one-to-one correspondence with multiedges in the corresponding solo diagram, we use the notations introduced in the beginning of Section 5.2 with multiedges as well as with solos), we have the following properties:

(i) if $x^{\mathtt{R}} \in e_1$ then all the other occurrences of x in e_1 and e_2 are E-occurrences in e_1: by (AC1), x cannot have another R-occurrence, moreover, if $x^{\mathtt{E}} \in e_2$ then, by (AC3), $s \lhd^{+} e_2$ thus e_2 is not a root of P, contradicting (AC2).

(ii) if $x^{\mathtt{R}} \in e_2$ then x has no other occurrence neither in e_1 nor in e_2: as for (i), x cannot have any other R-occurrence nor any E-occurrence in e_1 and finally, by (AC4), it cannot have an E-occurrence in e_2.

(iii) if $x^{\mathtt{E}} \in e_2$ then x has no R-occurrence neither in e_1 nor in e_2: otherwise we apply (i) or (ii).

We consider a connected component G_0 of $G_{\text{id}}[e_1, e_2]$. It is the underlying non-directed graph of the appropriate sub-graph G'_0 of G'. G'_0 is finite, G_0 is connected and, according to condition (AC1), each node has at most one incoming edge in G'_0. We just have to show that G'_0 has a root, *i.e.* that there is a name in P which has only E-occurrences in e_1 and e_2: if there is name with an E-occurrence in e_2 then this name cannot have any R-occurrence (by (iii)) and we are done, otherwise there is a name with a R-occurrence in e_2 and the corresponding occurrence in e_1 is an E-occurrence of a name x, if x has an R-occurrence then it is also in e_1 (by (ii)) and the corresponding occurrence in e_2 is an E-occurrence, a contradiction.

Finally by (AC5), G' has the property that only roots can be free nodes (according to the free/bound labeling of nodes in G). This property is preserved by contraction of identification edges thanks to (AC1). Moreover this property entails that at most one extremity of an identification edge can be free, and thus no freeness problem arises during contraction of identification edges.

Conclusion. We have been able to define a suitable restriction of the solos calculus that can be used as an intermediary step between the π-calculus and differential interaction nets. This led to a new proof of the bisimulation result presented in [EL08].

The technical choices in the design of our acyclic solos calculus were completely determined by the just mentioned goal. Nevertheless, it would be interesting to study acyclic solos for themselves. Their computational behaviour are in many ways similar to what happens in the π-calculus ("substitution-like" name passing in an "unification style" setting for example). It would be interesting to see if they contain specific communication primitives or if somehow the behaviour of an acyclic solos term always mimics the behaviour of a π-term.

References

[Ehr02] Thomas Ehrhard. On Köthe sequence spaces and linear logic. *Mathematical Structures in Computer Science*, 12(5):579–623, 2002.

[EL08] Thomas Ehrhard and Olivier Laurent. Interpreting a finitary pi-calculus in differential interaction nets. Long version of the CONCUR 07 paper, submitted to Information and Computation, available at `http://www.pps.jussieu.fr/~laurent/dinpilong.pdf`, April 2008.

[ER06] Thomas Ehrhard and Laurent Regnier. Differential interaction nets. *Theoretical Computer Science*, 364(2):166–195, 2006.

[Laf95] Yves Lafont. From proof nets to interaction nets. In Jean-Yves Girard, Yves Lafont, and Laurent Regnier, editors, *Advances in Linear Logic*, volume 222 of *London Mathematical Society Lecture Note Series*, pages 225–247. Cambridge University Press, 1995.

[LPV01] Cosimo Laneve, Joachim Parrow, and Björn Victor. Solo diagrams. In *Proceedings of the 4th conference on Theoretical Aspects of Computer Science, TACS'01*, number 2215 in Lecture Notes in Computer Science, pages 127–144. Springer-Verlag, 2001.

[LV03] Cosimo Laneve and Björn Victor. Solos in concert. *Mathematical Structures in Computer Science*, 13(5):657–683, 2003.

[MPW92] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes, I/II. *Information and Computation*, 100(1):1–77, 1992.

[PV98] Joachim Parrow and Björn Victor. The fusion calculus: Expressiveness and symmetry in mobile processes. In *Proceedings of the thirteenth annual symposium on Logic In Computer Science*, pages 176–185, Indianapolis, June 1998. IEEE, IEEE Computer Society Press.

We would like to thank to Cosimo Laneve for the helpful discussions we had about solos.