An Explicit Framework for Interaction Nets

Marc de Falco ⋆

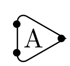
Institut de Mathématiques de Luminy
Université de la Méditerranée
Marseille, France

Abstract. Interaction nets are a graphical formalism inspired by Linear Logic proof-nets often used for studying higher order rewriting e.g. β-reduction. Traditional presentations of interaction nets are based on graph theory and rely on elementary properties of graph theory. We give here a more explicit presentation based on notions borrowed from Girard's *Geometry of Interaction*: interaction nets are presented as partial permutations and a composition of nets, the *gluing*, is derived from the execution formula. We then define contexts and reduction as the context closure of rules. We prove strong confluence of the reduction within our framework and show how interaction nets can be viewed as the quotient of some generalized proof-nets.

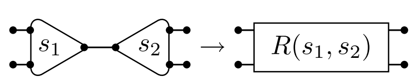
1 Introduction

Interaction nets were introduced by Yves Lafont in [Laf90] as a way to extract a model of computation from the well-behaved proof-nets of multiplicative linear logic. They have since been widely used as a formalism for the implementation of reduction strategies for the λ-calculus, providing an intuitive way to do explicit substitution [Mac98][MP98][Lip03].

Interaction nets are easy to present: a net is made of cells



with a fixed number of connection ports, depicted as big dots on the picture, one of which is distinguished and called the principal port of the cell, and of free ports, and of wires between those ports such that any port is linked by exactly one wire. Then we define reduction on nets by giving rules of the form

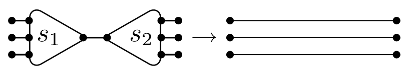


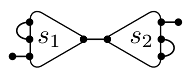
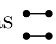
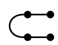
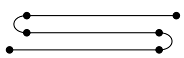
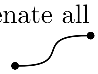
where the two cells in the left part are linked by their principal ports and the box in the right part is a net with the same free ports as the left part. Such a rule can be turned into a reduction of nets: as soon as a net contains the left part we replace it with the right part.

⋆ This work is supported by the French ANR project **CHoCo**

Even though this definition is sufficient to work with interaction nets, it is too limited to reason on things like paths or observational equivalence. One of the main issues comes from the fact that we do not really know what a net is. The situation is quite similar for graphs: we cannot study them relying using drawings only without being deceived by our intuition. Thus, we are inclined to give a precise definition of a graph as a binary relation or as a set of edges.

The main issue to give such a definition for interaction nets is that it should cope with reduction. As an example consider a graph-like construction over ports and a rule



Can it be applied to the interaction net  ? If we are rigorous the left part of the rule is not exactly contained in this net as  is not contained in  . Perhaps we could consider this last wire as composed of three smaller ones and two temporary ports like in  and the whole net after reduction would be  . But then, to get back a real interaction nets we would have to concatenate all those wires and erase the temporary ports, which would give us the net  . We will refer to this process of wire concatenation as *port fusion*.

There are many works giving definitions of interaction nets giving a rigorous description of reduction. Nevertheless, they all share a common point: they deal either implicitly or externally with port fusion. In the seminal article [Laf90] a definition of nets as terms with paired variables is given, it is further refined in [FM99]. In this framework an equivalence relation on variables deals with port fusion. In [Pin00] a concrete machine is given where the computation of the equivalence relation is broken into many steps. A rigorous approach sharing some tools with ours is given in [Vau07], port fusion is done there by an external port rewriting algorithm.

Therefore, we raise the following question: can we give a definition of interaction nets allowing a simple and rigorous description of reduction encompassing port fusion, and upon which we can prove results like strong confluence? This is the aim of this paper.

Our proposition is based on the following observation. When we plug the right part of a rule in a net, new wires are defined based on a back and forth process between the original net and this right part. Such kind of interaction is key to the *geometry of interaction* [Gir89] or *game semantics* [AJM94,HO00]. The untyped nature of interaction nets makes the former a possible way to express them. To be able to do so we need to express an interaction net as some kind of partial permutation and use a composition based on the so-called *execution formula*. Such presentation of multiplicative proof-nets has been made by Jean-Yves Girard in [Gir87]. If we try to think about the fundamental actions one needs to be able to do on interaction nets, it is quite clear that we can distinguish a *wire action* consisting in going from one port to another along a wire and the *cell action* consisting in going from one cell port to another inside the same cell. Those two actions lead to the description of a net as a pair of permutations. One might ask whether it is possible in some case to faithfully combine this pair in only one permutation, a solution to this question is what one could call a *geometry of interaction*.

The issue of port fusion is not inherent to interaction nets and can be found in other related frameworks. Diagram rewriting [Laf03] uses a well-behaved underlying category allowing mathematically the *straightening* of wires. But this is not free: the presentation is now vertically directed and lack the ease of definition of interaction nets for describing programs. Another work related to this problem is the presentation of multiplicative proof-nets by Hughes in [Hug05] where the author presents proof-nets as functions with a composition based on a categorical construction associated to traced monoidal categories [JSV96] which has been used to analyse Girard's *geometry of interaction* [AJ92,HS06]. A large part of our framework could be seen as a special case of a similar general categorical construction. Indeed, we are using the same tool as in those semantics, but our specialization to the partial injections of integers allows us to work on syntax and to stay in a completely untyped world.

2 Permutations and partial injections

We give here the main definitions and constructions that are going to be central to our realization of interaction nets. Those definitions are standard in the partial injections model of *geometry of interaction* [Gir87,DR95] or in the definition of the traced monoidal category PInj [HS06].

2.1 Permutations

We recall that a permutation of a set E is any bijection acting on E and we write $\mathfrak{S}(E)$ for the set of these permutations. For $\sigma \in \mathfrak{S}(E)$ we call *order* the least integer n such that $\sigma^n = id_E$, for $x \in E$ we write $\mathrm{Orb}_\sigma(x) = \{\sigma^i(x) \mid i \in \mathbb{N}\}$ and we call it the *orbit* of x, we write $\mathrm{Orbs}(\sigma)$ for the orbits of σ. If o is an orbit we write $|o|$ for its size.

We write $(c_1, \ldots, c_n)$ for the permutation sending c_i to c_{i+1}, for $i < n$, c_n to c_1 and being the identity elsewhere, we call it a *cycle* of *length* n which is also its order. Any permutation is a compound of disjoint cycles.

Let σ be a permutation of E and $\mathcal{L}$ any set, we say that σ is labelled by $\mathcal{L}$ if we have a function $l_\sigma : \mathrm{Orbs}(\sigma) \to \mathcal{L}$. We say that σ *has pointed orbits* if it is labelled by E and $\forall o \in \mathrm{Orbs}(\sigma)$ we have $l_\sigma(o) \in o$. Remark that an orbit is a sub-cycle and thus, having pointed orbits means that we have chosen a starting point in those sub-cycles.

2.2 Partial injections

A *partial injection (of integers)* f is a bijection from a subset $\mathrm{dom}(f)$ of $\mathbb{N}$, called its *domain*, to a subset $\mathrm{codom}(f)$ of $\mathbb{N}$, called its *codomain*. We write $f : A \twoheadrightarrow B$ to say that f is any partial injection such that $\mathrm{dom}(f) = A$ and $\mathrm{codom}(f) = B$. We write $f^\star$ for the inverse of this bijection viewed as a partial injection. We call *partial permutation* a partial injection f such that $\mathrm{dom}(f) = \mathrm{codom}(f)$.

2.3 Execution

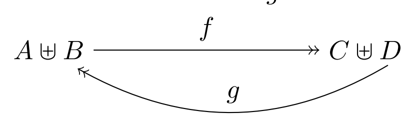
Let f be a partial injection and $E', F' \subseteq \mathbb{N}$. We write $f\restriction^{E'}_{F'}$ for the partial injection of domain $\{x \in E' \cap \mathrm{dom} f \mid f(x) \in F'\}$ ans such that $f\restriction^{E'}_{F'}(x) = f(x)$ where it is defined. We have

$$f\restriction^{E'}_{F'} : f^{-1}(F') \cap E' \twoheadrightarrow f(E') \cap F'$$

If $E = F$ and $E' = F'$ we write $f\restriction_{E'} = f\restriction^{E'}_{E'}$.

When $\mathrm{dom}(f) \cap \mathrm{dom}(g) = \emptyset$ and $\mathrm{codom}(f) \cap \mathrm{codom}(g) = \emptyset$, we say that f and g are *disjoint* and we define the sum $f + g$ and the associated refining order $\prec$ as expected. We have $\mathrm{dom}(f + g) = \mathrm{dom}(f) \uplus \mathrm{dom}(g)$ where $\uplus$ is the disjoint union.

Property 1 *Let $f : A \uplus B \twoheadrightarrow C \uplus D$ and $g : D \twoheadrightarrow B$ a situation depicted by the following diagram*



.

i For all $n \in \mathbb{N}$, the partial injection from A to C

$$\mathsf{Ex}_n(f, g) = f\restriction^A_C + (fgf)\restriction^A_C + \cdots + (f(gf)^n)\restriction^A_C$$

is well defined.

ii $(\mathsf{Ex}_n(f, g))_{n\in\mathbb{N}}$ is an increasing sequence of partial injections with respect to $\prec$, whose limit, the increasing union, is noted $\mathsf{Ex}(f, g)$.

iii If $\mathrm{dom}(f)$ is finite the sequence $(\mathsf{Ex}_n(f, g))_n$ is stationary and

$$\mathsf{Ex}(f, g) : A \twoheadrightarrow C$$

Fig. 1 gives a graphical presentation of execution.

Proof i) To assert the validity of the sum all we have to have show is that $\forall i \neq j \in \mathbb{N}$:

$$(f(gf)^i)(A) \cap (f(gf)^j)(A) \cap C = \emptyset$$
$$(f(gf)^i)^{-1}(C) \cap (f(gf)^j)^{-1}(C) \cap A = \emptyset$$

Suppose there is a $x \in (f(gf)^i)(A) \cap (f(gf)^j)(A) \cap C$, we set y and $z \in A$ such that $x = f(gf)^i(y) = f(gf)^j(z)$. We can further suppose that $i < j$, and we have $y = (gf)^{j-i}(z) \in B$, which is contradictory as $y \in A$ and $A \cap B = \emptyset$.

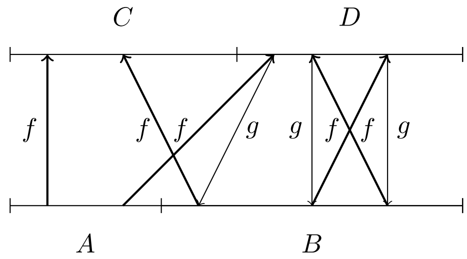


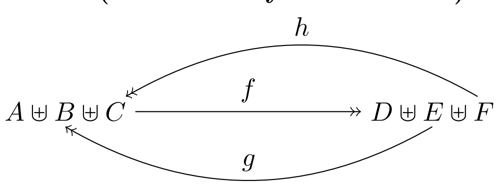
Fig. 1: Representation of $\mathsf{Ex}(f, g)$ with the notations of property 1

The other equality is proved in the same way.

ii) Let $n \leq m \in \mathbb{N}$ and $x \in \mathrm{dom}(\mathsf{Ex}_n(f, g))$, by definition of the sum there exists a unique k such that $\mathsf{Ex}_n(f, g)(x) = (f(gf)^k)(x)$. But then $x \in \mathrm{dom}(\mathsf{Ex}_m(f, g))$ and the uniqueness of k asserts that $\mathsf{Ex}_m(f, g)(x) = (f(gf)^k)(x)$. Thus, $\mathsf{Ex}_m(f, g)$ is a refinement of $\mathsf{Ex}_n(f, g)$.

iii) Suppose there is a $x \in A - \mathrm{dom}(\mathsf{Ex}(f, g))$, then we should have for all k, $(f(gf)^k)(x) \in D$ or else $\mathsf{Ex}(f, g)(x)$ would be defined. But D being finite, there exists $n \leq m$ such that $(f(gf)^n)(x) = (f(gf)^m)(x)$ and we get $x = (gf)^{m-n}(x) \in B$ which is contradictory. A simple argument on cardinal show then that $\mathrm{codom}(\mathsf{Ex}(f, g)) = C$. $\square$

Theorem 2 (Associativity of execution)

Let 

be three partial injections. We have $\forall n \in \mathbb{N}$

$$\mathsf{Ex}_n(\mathsf{Ex}_n(f, g), h) = \mathsf{Ex}_n(f, g + h) = \mathsf{Ex}_n(\mathsf{Ex}_n(f, h), g)$$

and thus

$$\mathsf{Ex}(\mathsf{Ex}(f, g), h) = \mathsf{Ex}(f, g + h) = \mathsf{Ex}(\mathsf{Ex}(f, h), g)$$

Proof Let $p \in \mathrm{dom}(\mathsf{Ex}_n(f, g + h))$, there exists $m \leq n \in \mathbb{N}$ such that

$$\begin{aligned}
\mathsf{Ex}_n(f, g + h)(p) &= f((g + h)f)^m(p) \\
&= (f(gf)^{i_1})h \ldots h(f(gf)^{i_k})(p) \text{ with } i_1 + \cdots + i_k + k - 1 = m \\
&= (\mathsf{Ex}_n(f, g)h\mathsf{Ex}_n(f, g) \ldots h\mathsf{Ex}_n(f, g))(p) \\
&= (\mathsf{Ex}_n(f, g)(h\mathsf{Ex}_n(f, g))^{k-1})(p) \\
&= \mathsf{Ex}_n(\mathsf{Ex}_n(f, g), h)(p)
\end{aligned}$$

By commutativity of $+$ we get the other equality. These equalities are directly transmitted to Ex. $\square$

This theorem is of utter significance, it is a completely localized version of Church-Rosser property. Indeed, we will see later that confluence results are corollary of this theorem.

2.4 w-permutations and Ex-composition

We call *w-permutation* an involutive partial permutation of finite domain.

Let σ and τ be disjoint w-permutations and let f be a partial injection with $\mathrm{dom}(f) \subseteq \mathrm{dom}(\sigma)$ and $\mathrm{codom}(f) \subseteq \mathrm{dom}(\tau)$. We call the Ex_0-*composition* of σ and τ along f the partial permutation

$$\sigma \overset{f}{\leftrightsquigarrow}_0 \tau = \mathsf{Ex}(\sigma + \tau, f + f^\star)$$

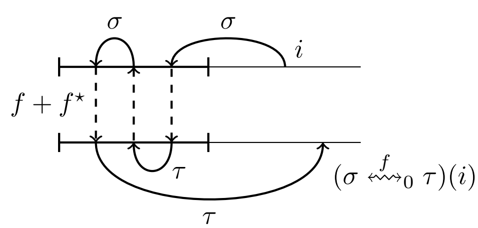


Fig. 2: Representation of the Ex_0-composition $\sigma \overset{f}{\leftrightsquigarrow}_0 \tau$

Fig. 2 gives a representation of this composition.

Property 3 *$\sigma \overset{f}{\leftrightsquigarrow}_0 \tau$ is a w-permutation.*

Proof Let x be an element of $\mathrm{dom}(\sigma + \tau)$, there exists n such that $(\sigma \overset{f}{\leftrightsquigarrow}_0 \tau)(x) = (f + f^\star)[(\sigma + \tau)(f + f^\star)]^n(x)$. Note that $(f + f^\star)^\star = f + f^\star$ and $(\sigma+\tau)^\star = \sigma+\tau$, and thus, we have $((f+f^\star)[(\sigma+\tau)(f+f^\star)]^n)^\star = [(f+f^\star)(\sigma+\tau)]^n(f+f^\star) = (f+f^\star)[(\sigma+\tau)(f+f^\star)]^n$. So $(\sigma \overset{f}{\leftrightsquigarrow}_0 \tau)^2(x) = x$. □

To define the final Ex-composition we want to recover the fix-points hidden by Ex in order to get the usual notion of *loops*. Suppose that there is an x_0 such that:

$$x_0 \xrightarrow{\sigma+\tau} y_0 \xrightarrow{f+f^\star} x_1 \ldots \xrightarrow{\sigma+\tau} y_n \xrightarrow{f+f^\star} x_n = x_0$$

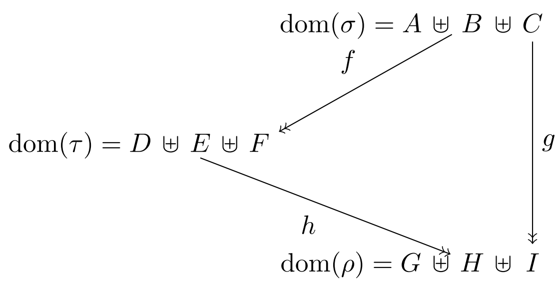
Everything being involutive, this loop is reversible and we get a new loop

$$y_0 \xleftarrow{f+f^\star} x_1 \ldots \xleftarrow{\sigma+\tau} y_n \xleftarrow{f+f^\star} x_0 \xleftarrow{\sigma+\tau} y_0$$

We say that the set $\{x_0, y_0, \ldots, x_{n-1}, y_n\}$ forms a *double orbit*, and we it can be fully reconstructed from any of its element. Therefore, recall that all this points are integers, and let R be a set comprised of the least element of each double orbit. We can define the Ex-composition, written $\sigma \overset{f}{\leftrightsquigarrow} \tau$, extending $\sigma \overset{f}{\leftrightsquigarrow}_0 \tau$ on R by $(\sigma \overset{f}{\leftrightsquigarrow} \tau)(r) = r$ for $r \in R$.

We can now give a direct corollary of theorem 2, stating some kind of associativity for the Ex-composition.

Corollary 4 *Let σ, τ, ρ be pairwise disjoint w-permutations with*



We have $\sigma \overset{f+g}{\leftrightsquigarrow} (\tau \overset{h}{\leftrightsquigarrow} \rho) = (\sigma \overset{f}{\leftrightsquigarrow} \tau) \overset{g+h}{\leftrightsquigarrow} \rho = (\sigma \overset{g}{\leftrightsquigarrow} \rho) \overset{f+h}{\leftrightsquigarrow} \tau$. When $h = 0$ we get $\sigma \overset{f+g}{\leftrightsquigarrow} (\tau + \rho) = (\sigma \overset{f}{\leftrightsquigarrow} \tau) \overset{g}{\leftrightsquigarrow} \rho = (\sigma \overset{g}{\leftrightsquigarrow} \rho) \overset{f}{\leftrightsquigarrow} \tau$.

3 The statics of interaction nets

We fix a countable set $\mathcal{S}$, whose elements are called *symbols*, and a function $\alpha : \mathcal{S} \to \mathbb{N}$, the *arity*. We will define nets atop $\mathbb{N}$ and in this context an integer will be called a *port*.

Definition 5 *An interaction net is an ordered pair $R = (\sigma_w, \sigma_c)$ where:*

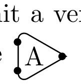
- *σ_w is a w-permutation. We write $P_l(R)$ for the fixed points of σ_w and $P(R)$ for the others.*
- *σ_c is a partial permutation of $P(R)$ with pointed orbits and labelled by $\mathcal{S}$ in such a way that $\forall o \in Orbs(\sigma_c), |o| = \alpha(l(o))$ where l is the labelling function.*

The elements of $P_l(R)$ are called *loops* and the other orbits of σ_w, which are necessarily of length 2, are called *wires*. The domain of σ_w is called the *carrier* of the net. We write $P_c(R) = \mathrm{dom}(\sigma_c)$, whose elements are called *cell ports*, and $P_f(R) = P(R) - P_c(R)$, whose elements are called *free ports*.

An orbit of σ_c is called a *cell*. We write pal for the pointing function of σ_w. Let c be a cell, $\mathrm{pal}(c)$ is its *principal port* and for $i < |c|$ the element $(\sigma_c^i \circ \mathrm{pal})(c)$ is its *ith auxiliary port*.

Note that a port is present in exactly one wire and at most one cell.

3.1 Representation

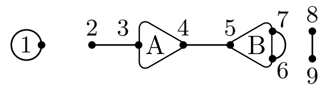
Nets admit a very natural representation. We shall draw a cell of symbol A as a triangle 

where the principal port is the dot on the apex and auxiliary ports are lined up on the opposing edge. We draw free ports as points. To finish the drawing we add a line between any two ports connected by a wire, and draw circles for loops.

As an example consider the net $R = (\sigma_w, \sigma_c)$ with

$$\sigma_w = (1)(2\ 3)(4\ 5)(6\ 7)(8\ 9) \text{ and } \sigma_c = (\dot{4}\ \ 3)_A(\dot{5}\ \ 6\ 7)_B$$

where permutations are given by cycle decomposition and $(\dot{c_1}\ \ c_2\ \dots\ c_n)_S$ is a cell of point c_1 and symbol S. This net will have the representation



3.2 Morphisms of nets and renaming

Definition 6 *Let $R = (\sigma_w, \sigma_c)$ and $R' = (\sigma'_w, \sigma'_c)$ be two interaction nets. The function $f : P(R) \mapsto P(R')$ is a* morphism from R to R' *iff*

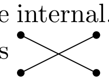
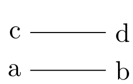
$$f \circ \sigma_w = \sigma'_w \circ f, \quad f(P_c(R)) \subseteq P_c(f(R')),$$

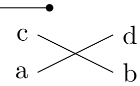
$$\forall p \in P_c(R), (f \circ \sigma_c)(p) = (\sigma'_c \circ f)(p),$$

and $\forall o \in Orbs(\sigma_c)$ we have $(f \circ pal)(o) = (pal \circ f)(o)$ and $l(o) = (l \circ f)(o)$. When f is the identity on $P_f(R)$ it is said to be an internal morphism.

Let us detail a bit more this definition. We note that for any two partial permutations σ and τ, the equation $f \circ \sigma = \tau \circ f$ induces that a $o \in \mathrm{Orbs}(\sigma)$ is mapped to an element $f(o) \in \mathrm{Orbs}(\tau)$ such that $|f(o)|$ is a divisor of $|o|$.

In this case a loop is sent to a loop, a wire to a loop or a wire, and a cell to another cell. The last two equations say that the principal port of cell is mapped to a principal port, and symbols are preserved. So a cell is mapped to a cell of same arity, and each port is mapped to the same type of port. Moreover only a wire linking free ports can be mapped to a loop or any kind of wire. As soon as the wire is linking one cell port the third condition on the morphism must send it to a wire of the same type.

With this remark, it is natural to call *renaming* (resp. *internal renaming*) an isomorphism (resp. internal isomorphism). An isomorphism class captures interaction nets as they are drawn on paper. On the other hand, an internal isomorphism class corresponds to interaction nets drawn where we have also given distinct names to free ports, hence the name internal. This is an important notion because the drawing  is the same as  Whereas the drawing 

is different from 

.

Remark 7 *Given the fact that nets have finite carriers we can always consider that two nets have disjoint carriers up to renaming.*

4 Tools of the trade

4.1 Gluing and cutting

Definition 8 *Let* $R = (\sigma_w, \sigma_c)$ *and* $R' = (\sigma'_w, \sigma'_c)$ *be two nets with disjoint carriers* [1] *and let* $f : P_f(R) \hookrightarrow P_f(R')$*. We call* gluing of R and R' along f *the net* $R \overset{f}{\leftrightsquigarrow} R' = (\sigma_w \overset{f}{\leftrightsquigarrow} \sigma'_w, \sigma_c + \sigma'_c)$*.*

From this definitions we get the following obvious facts:

$$\begin{aligned}
P(R \overset{f}{\leftrightsquigarrow} R') &= (P(R) - \mathrm{dom}(f)) \uplus (P(R') - \mathrm{codom}(f)) \\
P_c(R \overset{f}{\leftrightsquigarrow} R') &= P_c(R) \uplus P_c(R') \\
P_f(R \overset{f}{\leftrightsquigarrow} R') &= (P_f(R) - \mathrm{dom}(f)) \uplus (P_f(R') - \mathrm{codom}(f)) \\
R \overset{f}{\leftrightsquigarrow} R' &= R' \overset{f^\star}{\leftrightsquigarrow} R
\end{aligned}$$

For the special case of gluing where $f = 0$ we have $R \overset{0}{\leftrightsquigarrow} R' = (\sigma_w + \sigma'_w, \sigma_c + \sigma'_c)$, we write this special kind of gluing $R + R'$, it is the so-called parallel composition of the two nets. Fig. 3 gives a representation of gluing.

Property 9 *If* $R = R \overset{f}{\leftrightsquigarrow} R'$ *then* $f = 0$ *and* $R' = \mathbf{0} = (0, 0)$*. If* $\mathbf{0} = R \overset{f}{\leftrightsquigarrow} R'$ *then* $f = 0$ *and* $R = R' = \mathbf{0}$*.*

Proof We will only prove the first assertion, the second being similar. It is a direct consequence of the previous facts, R' must have no cells, no free ports and no loops. The only net having this property is the empty net $\mathbf{0}$. □

We can get some kind of associativity property for gluing.

Property 10 *Let* $R = (\sigma_w, \sigma_c)$*,* $S = (\tau_w, \tau_c)$ *and* $T = (\rho_w, \rho_c)$ *be nets of disjoint carriers and let* f, g *and* h *be partial injections satisfying the diagram of corollary 4 with respect to* σ_w, τ_w *and* ρ_w*.*

We have $R \overset{f+g}{\leftrightsquigarrow} (S \overset{h}{\leftrightsquigarrow} T) = (R \overset{f}{\leftrightsquigarrow} S) \overset{g+h}{\leftrightsquigarrow} T = (R \overset{g}{\leftrightsquigarrow} T) \overset{f+h}{\leftrightsquigarrow} S$*.*

Proof The wire part of the equality is a restriction of corollary 4 and the cell part is the associativity of $+$. □

The following corollary will often be sufficient.

Corollary 11 *If we have a decomposition* $R_0 = R \overset{f}{\leftrightsquigarrow} (S \overset{g}{\leftrightsquigarrow} T)$ *then there exists* f_S, f_T *such that* $R_0 = (R \overset{f_S}{\leftrightsquigarrow} S) \overset{g+f_T}{\leftrightsquigarrow} T$*.*

We can use the gluing to define dually the notion of cutting a subnet of an interaction net.

[1] Which is not a loss of generality thanks to remark 7.

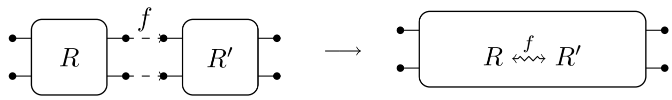


Fig. 3: Representation of the gluing of two interaction nets

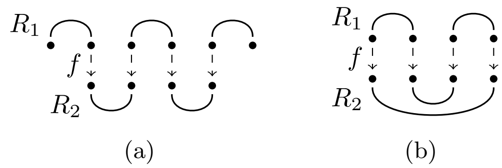


Fig. 4: Representation of two special cuttings: (a) a cutting of a single wire and (b) a cutting of a loop

Definition 12 *Let R be a net, we call* cutting of R *a triple* (R_1, f, R_2) *such that* $R = R_1 \overset{f}{\leftrightsquigarrow} R_2$*. Any net* R' *appearing in a cutting of* R *is called a* subnet of R*, noted* $R' \subseteq R$*.*

The Fig. 4 gives an example of cutting. The fact that we can cut many times a wire or that we can divide a loop in many wires hints at the complexity behind these definitions.

Property 13 *The relation $\subseteq$ is an ordering of nets.*

Proof The relation $\subseteq$ is **reflexive**: $R = R \overset{0}{\leftrightsquigarrow} \mathbf{0}$ and thus, $R \subseteq R$.

It is **antisymmetric**: let R_1 and R_2 be nets such that $R_1 \subseteq R_2$ and $R_2 \subseteq R_1$. We have $R_1 = R_2 \overset{f}{\leftrightsquigarrow} R'_2$ and $R_2 = R_1 \overset{g}{\leftrightsquigarrow} R'_1$.

So $R_2 = (R_2 \overset{f}{\leftrightsquigarrow} R'_2) \overset{g}{\leftrightsquigarrow} R'_1$. By applying the corollary 11 we get $R_2 = R_2 \overset{f_1}{\leftrightsquigarrow} (R'_2 \overset{g+f_2}{\leftrightsquigarrow} R'_1)$. and by applying the property 9 twice we get $R'_2 = R'_1 = \mathbf{0}$. So $R_1 = R_2$.

And it is **transitive**: let $R \subseteq S \subseteq T$, then $S = R \overset{f}{\leftrightsquigarrow} R'$ and $T = S \overset{g}{\leftrightsquigarrow} S'$, so $T = (R \overset{f}{\leftrightsquigarrow} R') \overset{g}{\leftrightsquigarrow} S'$. By applying the corollary 11 we have $T = R \overset{f_1}{\leftrightsquigarrow} (R' \overset{g+f_2}{\leftrightsquigarrow} S')$, that is to say $R \subseteq T$. □

4.2 Interfaces and contexts

To define reduction by using the subnet relation, it would be easier if we could refer implicitly to the identification function in a gluing. As an intuition, consider terms contexts with multiple holes, to substitute completely such contexts we could give a function from holes to terms and fill them accordingly. But a more natural definition would be to give a distinct number to each hole and to fill based on a list of terms. The substitution would give the first term to the first hole, and so on. The following definition is a direct transposition of this idea in the framework of interaction nets.

Definition 14 *We call* interface of a net R *a subset* $I = \{p_1, \dots, p_n\}$ *of* $P_f(R)$ *together with a linear ordering, the length of the order chain* $p_1 < \dots < p_n$ *is called the* size*. We say that* R *contains the interface* I*, noted* $I \subset R$*. An interface is* canonical *if it contains all the free ports of a net.*

Let I *and* I' *be disjoint interfaces of the same net, we write* II' *the union of these subsets ordered by the concatenation of the two order chains. Precisely* $x \leq_{II'} y \iff x \leq_I y$ *or* $x \leq_{I'} y$ *or* $x \in I \wedge y \in I'$*.*

Let I *and* I' *be two interfaces of same order, there exists one and only order-preserving bijection from* I *to* I' *that we write* $\rho(I, I')$ *and call the* chord between I and I'.

We call context *a couple* (R, I) *where* I *is an interface contained in the net* R*, it is written* R^I*.*

Let R^I *and* ${R'}^{I'}$ *be two contexts with interfaces of same order, we write*

$$R^I \leftrightsquigarrow {R'}^{I'} = R \overset{\rho(I,I')}{\leftrightsquigarrow} R'$$

In the following when we write $R^I \leftrightsquigarrow {R'}^{I'}$ we implicitly assume that I and I' are of same size.

We now can state commutativity of gluing directly, the proof being trivial.

Property 15 (Commutativity of gluing) $R^I \leftrightsquigarrow {R'}^{I'} = {R'}^{I'} \leftrightsquigarrow R^I$

The following trivial fact asserts that any gluing can be seen as a context gluing.

Fact 16 *Let* $R \overset{f}{\leftrightsquigarrow} R'$ *be a gluing, there exist interfaces* $I \subset R$ *and* $I' \subset R'$ *of same order such that* $R \overset{f}{\leftrightsquigarrow} R' = R^I \leftrightsquigarrow {R'}^{I'}$*.*

Corollary 17 $R_1 \subseteq R \iff \exists I_1, R_2, I_2$ *such that* $R = {R_1}^{I_1} \leftrightsquigarrow {R_2}^{I_2}$

We can now restate the corollary 11 with interfaces:

Corollary 18 *For all nets* R, S, T *and interfaces* I, J, K, L*, there exists interfaces* I', J', K', L' *such that*

$$R^I \leftrightsquigarrow \left(S^J \leftrightsquigarrow T^K\right)^L = \left(R^{I'} \leftrightsquigarrow S^{J'}\right)^{L'} \leftrightsquigarrow T^{K'}$$

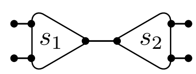
5 Dynamics

Definition 19 *Let s_1 and s_2 be symbols. We call* interaction rule for (s_1, s_2) *a couple* $({R_r}^{I_r}, {R_p}^{I_p})$ *where*

$$R_r = \begin{pmatrix} (b\ c)(a_1\ b_1)\ldots(a_n\ b_n)(c_1\ d_1)\ldots(c_m\ d_m), \\ (\overset{\bullet}{b}\ \ b_1\ \ \ldots\ \ b_n)_{s_1}(\overset{\bullet}{c}\ \ c_1\ \ \ldots\ \ c_m)_{s_2} \end{pmatrix}$$

and I_r and I_p are both canonical – comprised of all free ports – and of same size.

Let $\mathcal{R} = ({R_r}^{I_r}, {R_p}^{I_p})$ be a rule we call reduction by $\mathcal{R}$ *the binary relation $\xrightarrow{\mathcal{R}}$ on nets such that for all renaming α and β, and for all net S with $S = R^I \leftrightsquigarrow \alpha(R_r)^{\alpha(I_r)}$ we set $S \xrightarrow{\mathcal{R}} S'$ where $S' = R^I \leftrightsquigarrow \beta(R_p)^{\beta(I_p)}$.*

The net R_r has the representation  . Remark that the reduction is defined as soon as a net contains a renaming of the redex R_r. This reduction appears to be non-deterministic but it is only the expansion of a deterministic reduction to cope with all possible renamings.

Property 20 *Let R be a net and $\mathcal{R}_1$, $\mathcal{R}_2$ be two interaction rules applicable on R on distinct redexes such that $R_1 \xleftarrow{\mathcal{R}_1} R \xrightarrow{\mathcal{R}_2} R_2$ and all the ports both in R_1 and R_2 are also in R. There exists a net R' such that $R_1 \xrightarrow{\mathcal{R}_2} R' \xleftarrow{\mathcal{R}_1} R_2$.*

Proof For $i = 1, 2$, set $\mathcal{R}_i = ({R_{r,i}}^{I_{r,i}}, {R_{p,i}}^{I_{p,i}})$. The shape of redexes allow us to assert that if they are distinct then they are disjoint. As R contains both a redex $\alpha_1(R_{r,1})$ and a redex $\alpha_2(R_{r,2})$, then we can deduce that $\alpha_1(R_{r,1}) + \alpha_2(R_{r,2}) \subseteq R$. More precisely we have

$$R = (\alpha_1(R_{r,1}) + \alpha_2(R_{r,2}))^{\alpha_1(I_{r,1})\alpha_2(I_{r,2})} \leftrightsquigarrow {R_0}^I$$

We get

$$R_1 = (\beta_1(R_{p,1}) + \alpha_2(R_{r,2}))^{\beta_1(I_{p,1})\alpha_2(I_{r,2})} \leftrightsquigarrow {R_0}^I$$

for a renaming β_1, and the same kind of expression for R_2. It is straightforward to check that the net

$$R' = (\beta_1(R_{p,1}) + \beta_2(R_{p,2}))^{\beta_1(I_{p,1})\beta_2(I_{p,2})} \leftrightsquigarrow {R_0}^I$$

satisfies the conclusion by applying property 10. The very existence of this net relies on the disjointness of the $\beta_i(R_{p,i})$ which is ensured by the hypothesis on ports contained in both R_1 and R_2. □

Corollary 21 *Let $\mathcal{L}$ be a set of rules such that for any pair of symbols there is at most one rule over them. The reduction $\xrightarrow{\mathcal{L}} = \bigcup_{\mathcal{R}\in\mathcal{L}} \xrightarrow{\mathcal{R}}$ is strongly confluent up to a renaming.*

By *up to a renaming* we mean that we might have to rename one of the nets in a critical pair before joining them. This is due to the disjointness condition in property 20. Remark that we can always substitute one of the branch of the critical pair by another instance of the same rule on the same redex in such a way that this condition is ensured.

6 Interaction nets are the Ex-collapse of Axiom/Cut nets

We introduce now a notion of nets lying between proof-nets of multiplicative linear logic and interaction nets. When we plug directly two interaction nets a complex process of wire simplification occurs. When we plug two proof-nets we only add special wires called *cuts* and we have an external notion of reduction performing such simplification. In this section we define nets with two kinds of wires: *axioms* and *cuts*. Those nets allow us to give a precise account of the folklore assertion that interaction nets are a quotient of multiplicative proof-nets.

6.1 Definition and juxtaposition

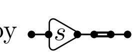
Definition 22 *An* Axiom/Cut net*,* AC net *for short, is a tuple* $R = (\sigma_A, \sigma_C, \sigma_c)$ *where:*

- σ_A *and* σ_C *are w-permutations of finite domain included in* $\mathbb{P}$*, such that* $\mathrm{dom}(\sigma_C) \subseteq \mathrm{dom}(\sigma_A)$*,* σ_C *has no fixed points and if* $(a\ b)$ *is an orbit of* σ_C *then there exists* $c \neq a$ *and* $d \neq b$ *such that* $(c\ a)$ *and* $(b\ d)$ *are orbits of* σ_A *We write* $P_l(R)$ *for the fixed points of* σ_A *and* $P(R) = \mathrm{dom}(\sigma_A) - \mathrm{dom}(\sigma_C) - P_l(R)$*.*
- σ_c *is an element of* $\mathfrak{S}(P_c(R))$*, where* $P_c(R) \subseteq P(R)$*, has pointed orbits and is labelled by* $\mathcal{S}$ *in such a way that* $\forall o \in Orbs(\sigma_c), |o| = \alpha(l(o))$ *where* l *is the labelling function.*

The orbits of σ_C, called *cuts*, are some kind of undirected unary cells linking orbits of σ_A, called *axioms*.

We directly adapt the representation of interaction nets to AC nets by displaying σ_c as double edges. For example the AC net $R = (\sigma_A, \sigma_C, \sigma_c)$ with

$$\sigma_A = (1\ 2)(3\ 4)(5\ 6), \sigma_C = (2\ 3), \sigma_c = (\dot{4}\ \ 5)_s$$

will be represented by  .

We can adapt most of the previous definitions for those nets, most importantly free ports, interfaces and contexts. The nice thing about AC nets is that they yield a very simple composition.

Definition 23 *Let* $R^I = (\sigma_A, \sigma_C, \sigma_c)$ *and* ${R'}^{I'} = (\tau_A, \tau_C, \tau_c)$ *be two contexts on AC nets with disjoint carriers, with* $I = i_1 > \cdots > i_n$ *and* $I' = i'_1 > \cdots > i'_n$*.*

We call juxtaposition of R^I and ${R'}^{I'}$ *the AC net*

$$R^I \leftrightarrow {R'}^{I'} = (\sigma_A + \tau_A, \sigma_C + \tau_C + (i_1\ i'_1) \ldots (i_n\ i'_n), \sigma_c + \tau_c)$$

The juxtaposition is from the logical point of view a generalized cut, and its interpretation in terms of permutation is exactly the definition made by Jean-Yves Girard in [Gir87].

6.2 Ex-collapse

Property 24 *Let $R = (\sigma_A, \sigma_C, \sigma_c)$ be an AC net and $f : \mathbb{P} \hookrightarrow \mathbb{P}$ be such that* $\mathrm{dom}(\sigma_C) = \mathrm{dom}(f)$ *and* $\mathrm{codom}(f) \cap \mathrm{dom}(\sigma_A) = \emptyset$.

The couple $(\sigma_A \overset{f}{\leftrightsquigarrow} f \circ \sigma_C \circ f^\star, \sigma_c)$, is an interaction net.

It does not depend on f and we call it the Ex-collapse *of R, noted* $\mathsf{Ex}(R)$.

For the definition of the Ex-composition to be correct, we have to delocalize σ_C to a domain disjoint from $\mathrm{dom}(\sigma_A)$. The Ex-collapse amounts to replace any maximal chain $a_1 \xrightarrow{\sigma_A} b_1 \xrightarrow{\sigma_C} a_2 \ldots b_{n-1} \xrightarrow{\sigma_A} a_n$ by a chain $a_1 \xrightarrow{\sigma_A} b_1 \xrightarrow{f} f(b_1) \xrightarrow{f \circ \sigma_C \circ f^\star} f(a_2) \xrightarrow{f^\star} \ldots b_{n-1} \xrightarrow{\sigma_A} a_n$ and then to compute the Ex-composition to get $a_1 \xrightarrow{\sigma_A \overset{f}{\leftrightsquigarrow} \sigma_C} a_n$.

Proof It comes from the definitions of the Ex-composition and from property 3. □

Property 25 *For each interaction net R there exists a unique AC net R' of the form $(\sigma_A, 0, \sigma_c)$ such that $\mathsf{Ex}(R') = R$. R' is said to be* cutfree.

Proof If $R = (\tau_w, \tau_c)$ we only have to take $R' = (\tau_w, 0, \tau_c)$. Uniqueness comes from the fact that $\sigma \overset{0}{\leftrightsquigarrow} 0 = \sigma$. □

Definition 26 *Let R and R' be two AC nets, we say that* R and R' are Ex-equivalent*, noted $R \overset{\leftrightsquigarrow}{\sim} R'$ when $\mathsf{Ex}(R) = \mathsf{Ex}(R')$.*

We have an obvious correspondence between juxtaposition and gluing.

Property 27 $\mathsf{Ex}(R^I \leftrightarrow R'^{I'}) = \mathsf{Ex}(R)^I \leftrightsquigarrow \mathsf{Ex}(R')^{I'}$

Therefore we can claim that

Interaction nets are the quotient of AC nets by $\overset{\leftrightsquigarrow}{\sim}$.

Conclusion

We could not include all of the possible extensions of our framework in this paper. Most of this results can be found in [dF09]. We have: 1) a double-pushout approach to reduction by mean of the category of interaction nets and morphisms (2) a rigorous definition of boxes as a partial labelling of cells (3) definitions of paths in a net, path reduction and proofs that the path reduction is strongly confluent (4) a full implementation in Haskell.

References

AJ92. S. Abramsky and J. Jagadeesan. New foundations for the geometry of interaction. In *Logic in Computer Science, 1992. LICS'92., Proceedings of the Seventh Annual IEEE Symposium on*, pages 211–222, 1992.

AJM94. S. Abramsky, R. Jagadeesan, and P. Malacaria. Full abstraction for PCF (extended abstract). In Masami Hagiya and John C. Mitchell, editors, *Theoretical Aspects of Computer Software. International Symposium TACS'94*, number 789 in Lecture Notes in Computer Science, pages 1–15, Sendai, Japan, April 1994. Springer-Verlag.

dF09. M. de Falco. *Géométrie de l'Interaction et Réseaux d'Interaction Différentiels*. Thèse de doctorat, Université d'Aix-Marseille 2, 2009.

DR95. Vincent Danos and Laurent Regnier. Proof-nets and the Hilbert space. In Jean-Yves Girard, Yves Lafont, and Laurent Regnier, editors, *Advances in Linear Logic*, volume 222 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, 1995.

FM99. M. Fernandez and I. Mackie. A calculus for interaction nets. *Lecture Notes in Computer Science*, 1702:170–187, 1999.

Gir87. J.-Y. Girard. Multiplicatives. In Lolli, editor, *Logic and Computer Science : New Trends and Applications*, pages 11–34, Torino, 1987. Università di Torino. Rendiconti del seminario matematico dell'università e politecnico di Torino, special issue 1987.

Gir89. J.-Y. Girard. Geometry of interaction I: an interpretation of system F. In Valentini Ferro, Bonotto and Zanardo, editors, *Proceedings of the Logic Colloquium 88*, pages 221–260, Padova, 1989. North-Holland.

HO00. M. Hyland and L. Ong. On full abstraction for PCF. *Information and Computation*, 163:285–408, 2000.

HS06. E. Haghverdi and P. Scott. A categorical model for the geometry of interaction. *Theoretical Computer Science*, 350(2-3):252–274, 2006.

Hug05. D. Hughes. Simple multiplicative proof nets with units. Archived as math.LO/0507003 at arXiv.org. Submitted., March 2005.

JSV96. A. Joyal, R. Street, and D. Verity. Traced Monoidal Categories. In *Proc. Comb. Phil. Soc*, volume 119, pages 447–468, 1996.

Laf90. Y. Lafont. Interaction nets. In *Proceedings of the 17^{th} Annual ACM Symposium on Principles of Programming Languages*, pages 95–108, San Francisco, 1990. Association for Computing Machinery, ACM Press.

Laf03. Y. Lafont. Towards an Algebraic Theory of Boolean Circuits. *Journal of Pure and Applied Algebra*, 184(2-3):257–310, 2003.

Lip03. S. Lippi. Encoding left reduction in the λ-calculus with interaction nets. *Mathematical Structures in Computer Science*, 12(06):797–822, 2003.

Mac98. I. Mackie. YALE: yet another lambda evaluator based on interaction nets. In *Proceedings of the third ACM SIGPLAN international conference on Functional programming*, pages 117–128. ACM New York, NY, USA, 1998.

MP98. I. Mackie and J. S. Pinto. Compiling the Lambda Calculus into Interaction Combinators. In *Logical Abstract Machines workshop*, 1998.

Pin00. J. S. Pinto. Sequential and concurrent abstract machines for interaction nets. In *FOSSACS '00: Proceedings of the Third International Conference on Foundations of Software Science and Computation Structures*, pages 267–282. Springer-Verlag London, UK, 2000.

Vau07. L. Vaux. *Lambda-calcul différentiel et logique classique*. Thèse de doctorat, Université de la Méditerranée, 2007.