Rewritings in Polarized (Partial) Proof Structures*

Christophe Fouqueré and Virgile Mogbil

LIPN – UMR7030
CNRS – Université Paris 13
99 av. J-B Clément, F–93430 Villetaneuse, France
{christophe.fouquere,virgile.mogbil}@lipn.univ-paris13.fr

Abstract. This paper is a first step towards a study for a concurrent construction of proof-nets in the framework of linear logic after Andreoli's works, by taking care of the properties of the structures. We limit here to multiplicative linear logic. We first give a criterion for closed modules (i.e. validity of polarized proof structures), then extend it to open modules (i.e. validity of partial proof structures) distinguishing criteria for acyclicity and connectability. The keypoint is an extensive use of the fundamental structural properties of the logics. We consider proof structures as built from n-ary bipolar objects and we show that strongly confluent (local) reductions on such objects are an elegant answer to the correctness problem. This has natural applications in (concurrent) logic programming.

1 Introduction

Girard in his seminal paper [8] gave a parallel syntax for multiplicative linear logic as oriented graphs called *proof-nets*. A *correctness criterion* enables one to distinguish sequentializable proof-structures (the so called proof-nets) from "bad" structures. After Girard's long trip correctness criterion, numerous equivalent properties were found. In particular, Danos and Regnier [7] proved that *switched* proof-structures should be trees. Furthermore, Danos implemented the criterion by means of a contraction relation on proof structures: binary connectives are the main elementary objects of the structures and are reduced by the relation. While a lot of research has been done on such correctness criteria, it still remains to study sequentialization of *polarized* as well as *partial* proof-structures. We generalize in this paper Danos and Regnier results to these two cases and show that the framework of proof-net rewritings leads to elegant results. In our case, structures are built from n-ary bipolar objects and we show that a strongly confluent (local) reduction may be defined as these elementary objects really take care of fundamental properties.

Such structures arise naturally after Andreoli's works [2–4] in logic programming: after showing in [1] that linear logic, a resource-conscious logic, may be used as a programming language[1] using a standard, sequential approach, he switches to a proof-net presentation as this syntax affords a desequentialized presentation of proofs, hence a

* Partially supported by ACI NIM project Géométrie du Calcul (GEOCAL), France.

[1] Full first-order linear logic can be used as a programming language. However, we restrict in this paper to propositional multiplicative linear logic.

concurrent way to compute them at the expense of a correctness criterion that guarantees to recover sequentialization, i.e. validity of proofs.

In this paper, we search for a generalization of Andreoli's results in order to have full expressivity. For that purpose, we depart from his approach by adopting a graph point of view. *Modules*, as graph elements, arise naturally from proof nets. In a few words, associativity, commutativity and focalization lead to polarize formulae, hence to stratify proofnets.[2] It turns out that polarization may enhance proof search, hence is central to prove that full linear logic could be a logic programming language. This fundamental notion was later considered in Girard's works [9], and also in Laurent's works about Polarized Linear Logic (LLP). Consequently our basic objects are proof structures with two strata we call *bipolar structures*: bipolarity is a key tool to get a rewriting system for checking correctness (2). Bipolar structures become computational structures as composition of such structures corresponds to some kind of progression rule in logic programming. As we shall show in the next sections, applying such a rule is nothing more than a composition of partial proof structures whose correctness is stated locally.

Andreoli set up this desequentialized framework for middleware infrastructures. In such applications, software agents must satisfy requests or goals by executing concurrently actions on a shared environment: actions transform the environment by deleting resources and creating new sets of results. Andreoli focused on *transitory* proof-structures, i.e. actions always create new resources. Moreover, he imposes prerequisites of actions to be satisfied in order to execute them: the proof construction is done bottom-up. As we shall see, these two hypotheses greatly simplify the problem of defining formally conditions under which actions may be undertaken. On the contrary, we constrain neither the structure of modules, nor the application order. It is then possible to define actions that kill resources or to anticipate consequences of resources still to be acquired. Furthermore, we depart from Andreoli's approach for defining a correctness criterion. His method is based on a computation of domination forests in the spirit of Murawski and Ong's approach [14]. We adopt here a completely different strategy. We define reduction relations in order to get the correctness property.

The following section gives basic definitions. We formally present modules from elementary ones, graphically and in terms of formulae. We specify in which sense a module is correct, i.e. computation is allowed. Section 3 is devoted to closed modules. A closed module is equivalent to a proof structure. Although closed modules are an extreme special case of modules, the methodology we use introduces naturally the way we consider open modules. In a first attempt, in the spirit of the resolution rule in logic programming we define a rewriting rule on modules: a transformation of a module may be viewed as a deconstruction of the proof structure. Correct normal forms are easily characterized. Extending the Danos-Regnier criterion,[3] we deduce a correctness criterion for closed modules as our rewriting rule and its inverse are stable wrt connectedness and acyclicity. We define next a modified version of the previous rewriting system: using polarization and focalization, the reduction becomes fully local: each step reduces one elementary object of our system without any global condition. Open

[2] Distributivity contributes to it when dealing with the additive part of linear logic.

[3] The Danos-Regnier criterion is based on graph properties of proof nets: correct proof structures, i.e. proof nets, are in some sense the connected and acyclic ones.

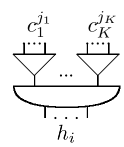
modules, i.e. modules without constraints, are studied in section 4. We prove that the Danos-Regnier criterion may be extended to open modules replacing the connectedness by a *connectability* property. We give two rewriting systems as acyclicity and connectability differ fundamentally. These two systems may be viewed as variations over the one we give for closed modules. We end with a study on incrementality wrt composition of modules. In terms of computation, elementary modules compete to modify some current open module (the environment): actions are concurrent when two such elementary modules are composed in disjoint parts of the environment. It is then crucial to be able to define rewriting systems that commute with composition. We show that we have to restrict previous rewriting systems for that purpose. However, the rewriting systems have to be split into two parts: one commutes with composition, the other is a post-treatment necessary to test correctness of composition.[4]

2 Basic definitions

Elementary bipolar modules are our basic blocks. They are interpreted as elementary actions that can take place during an execution. In terms of graph, applying an action is represented as a wire, i.e. composition, of the corresponding (elementary) module onto the current graph.

Definition 1. *An* elementary bipolar module *(EBM)* M *is given by a finite set* $\mathcal{H}(M)$ *of propositional variables (called* hypotheses*)* h_i *and a non empty finite set* $\mathcal{C}(M)$ *varying over* k *of finite sets of propositional variables (called* conclusions*)* c_k^j*. Variables are supposed pairwise distinct.*[5] *The set of propositional variables appearing in* M *is noted* $v(M)$*. Equivalently, one can define it as an oriented graph with labelled pending links and one positive pole under a finite set of negative poles. Its type* $t(M)$ *and drawing are given in the following way:*

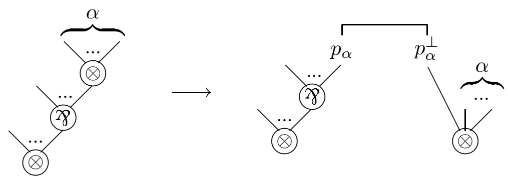
$$t(M) = (\bigotimes_i h_i) \multimap (\mathcal{⅋}_k (\bigotimes_{j_k} c_k^{j_k}))$$



Informally, the EBM has the following operational bottom-up reading: being given in some context a multiset of hypotheses (i.e. their tensor), this one is replaced by ($\multimap$) each of the mutisets of conclusion, these last have to be used in separate contexts (⅋ is the logical dual of $\otimes$). This specification of modules comes from the fact that connectives are naturally split into two sets: e.g. $\otimes$ is said positive, while ⅋ is negative. Propositional variables are declared positive, and their negation negative. Formulae alternate positive and negative levels up to propositional variables. Note that we use conveniently a two-sided style for formula and sequent presentations, even if our basic objects are proofnets. It is in fact possible to flatten proofnets to get bipolar structures related by links on fresh variables:

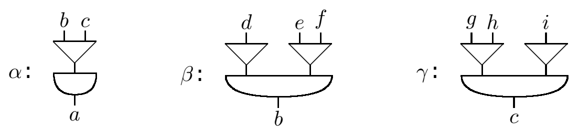
[4] Complements and some technical proofs are available in http://xxx.lanl.gov/cs/0411029.

[5] This restriction is taken for simplicity. The framework can be generalized if we consider multisets (of hypotheses and conclusions) instead of sets, and add as required a renaming mechanism: the results in this paper are still true.



If we notice that a variable and its negation cannot be together linked to negative nodes (it would contradict the correctness criterion), we can always suppose that, say, positive variables are linked to negative nodes. Finally, it may be the case that some bipolar structure (thus beginning with a positive node at bottom) has no negative variable: add then the constant $\mathbf{1}$, neutral for $\otimes$. Allowing abusively unary $\otimes$ and $⅋$ connectives, these (elementary) bipolar structures are the clauses of our programming language. We thus conveniently suppose that $⅋_k F_k = \bigotimes_k F_k = F_1$ when the domain of k is of cardinal 1. Moreover, if the domain of i is empty, $(\bigotimes_i h_i) \multimap C = \mathbf{1} \multimap C$ and if the domain of j_k for some k is empty $(\bigotimes_{j_k} c_k^{j_k}) = \bot$.

Example 1. The EBMs α, β and γ of respective types $t(\alpha) = a \multimap (b \otimes c)$, $t(\beta) = b \multimap (d ⅋ (e \otimes f))$ and $t(\gamma) = c \multimap ((g \otimes h) ⅋ i)$ are drawn in the following way:



Three kinds of EBMs are of special interest: An EBM is *initial* (resp. *final*) iff its set of hypotheses is empty (resp. its set of conclusions is empty). An EBM is *transitory* iff it is neither initial nor final. Initial EBMs allow to declare available resources, though final EBMs stop part of a computation by withdrawing a whole set of resources. Transitory EBMs can be seen as definite clauses in standard logic programming. Roughly speaking, a (bipolar) module (BM) is a set of EBMs such that a label appears at most once as a conclusion and at most once as a hypothesis. A label appears as a conclusion and as a hypothesis when two EBMs are linked by this label. As we search for correctness criteria wrt composition of modules (i.e. execution of the program), we give below an inductive definition of bipolar modules.

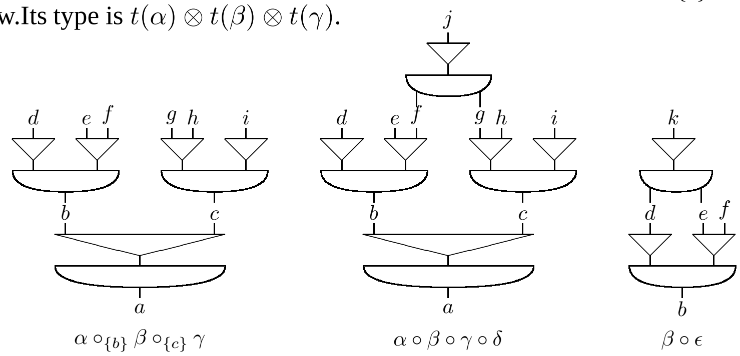
Definition 2 (BM). *A* bipolar module *(BM) is defined inductively in the following way:*

- *An EBM is a BM.*
- *Let M and N be a BM, let $I = (\mathcal{C}(M) \cap \mathcal{H}(N)) \cup (\mathcal{H}(M) \cap \mathcal{C}(N))$, their* composition *wrt the* interface *I, $M \circ_I N$ is a BM with :*
  - *the set of hypothesis (resp. conclusions) $\mathcal{H}(M \circ_I N)$ is the hypothesis (resp. conclusions) of M and N which are not in I*
  - *$t(M \circ_I N) = t(M) \otimes t(N)$ and $v(M \circ_I N) = v(M) \cup v(N)$.*

The border *$b(M)$ of a BM M is the union of the hypotheses and the conclusions.*

The informal explanation given before is more general than this definition because we define BM incrementally. However, we abusively do not consider these differences in the following as properties will be proven in the general case. The interface will be omitted when it is clear from the context. Note that the interface may be empty: it only means that two computations are concurrently undertaken, currently without any shared resources. A BM may not correspond to a valid computation: e.g. we do not want to accept that some action uses two resources in disjunctive situation! Correctness has obviously to be defined wrt the underlying Linear Logic as we do below. Finally, note that when a BM is correct, it represents the history of the computation whereas its conclusion is the current available environment.

Example 2. The composition of the EBMs α, β and γ is the BM $\alpha \circ_{\{b\}} \beta \circ_{\{c\}} \gamma$ drawn below.Its type is $t(\alpha) \otimes t(\beta) \otimes t(\gamma)$.



Definition 3 (Correctness (wrt sequentialization)). *Let M be a BM, M is* correct *iff there exists a formula C built with the connectives $\otimes$ and $\parr$, and the variables $\mathcal{C}(M)$ such that the sequent $\mathcal{H}(M), t(M) \vdash C$ is provable in Linear Logic.*

Example 3. Let us give two more BMs δ and ϵ of respective types $(f \otimes g) \multimap j$ and $(d \otimes e) \multimap k$.

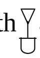
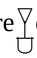
– The following sequent is provable in LL: $a, t(\alpha \circ \beta \circ \gamma \circ \delta) \vdash d \parr (e \otimes j \otimes h) \parr i$. The (correct) BM $\alpha \circ \beta \circ \gamma \circ \delta$ is drawn in the previous figure.
– The BM $\beta \circ \epsilon$ is not correct: there is a cycle through d and e.

As we shall focus first on characterizing correctness on closed modules, and then generalize our results to open modules, we adjoin to the term *correct* the kind of modules we speak of, e.g. c-correct when the module is closed, o-correct when it is open.

3 Closed modules

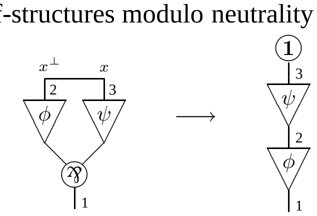
A closed module is a BM where the sets of hypotheses and conclusions are empty. Correctness of closed modules may be tested either in sequent calculus or by means of (simple oriented) graphs called *proof-nets*. We use this latest representation in this section. A *correctness criterion* enables one to distinguish sequentializable proof-structures (say such oriented graphs) from "bad" structures. The reader may find in [7] the definitions of proof structures and switchings. One generalizes this definition to n-ary connectives in the obvious way (taking care of associativity and commutativity of $\otimes$ and $\parr$) in place of standard binary ones. One modifies in the same way the definitions of switching introducing generalized switches. In particular a n-ary $\parr$ connective has n switched positions. One still can define switched proof-structures and a criterion generalizing Danos-Regnier correctness criterion: A closed module M is DR-correct iff for all generalized switches s on M^o, $s(M^o)$ is acyclic and connected, where M^o is the proof structure associated to $t(M)^{\perp}$.[6] We immediately have the following proposition as a corollary of the DR-criterion theorem (remember that a c-correct module is a correct closed module):

Proposition 1 (c-correction). *Let M be a closed module, M is c-correct iff $t(M) \vdash$ is provable in Linear Logic, iff M is DR-correct.*

Remember that the equivalent (binary) Danos correctness criterion may be implemented by means of a contraction relation on proof structures. However, intermediate reduced structures may not be describable in terms of (bipolar) modules. Moreover such a contraction relation does not take advantage of the incremental definition of modules as a composition of elementary bipolar modules. A first idea consists of representing the resolution step (implicit in EBMs composition) in terms of modules. We first give below such a (small step) reduction rule that is stable wrt correctness with  as the correct normal form, where  denotes the terminal EBM (i.e. smallest final and initial). We give then a second proposal that takes care of the focalization property. Though a resolution step reduces one variable, this second formulation uses as a whole the structure of a module thanks to focalization. The focalization property states that a sequent is provable iff there exists a proof s.t. decomposition of the positive stratum of formulae is done in one step. Considering bipolar modules, it means that one may define a reduction relation s.t. each step reduces one positive-negative pair of nodes.

Let $\rightsquigarrow^*_{\Theta}$ be the transitive closure of the following relation defined on literals of a proof-structure Θ: let u and v be two literals of Θ, $u \rightsquigarrow_{\Theta} v$ iff $u^{\perp}$ and v are in the same subtree with root $\otimes$ of the formula corresponding to Θ. We note $u \rightsquigarrow^* v$ when there is no ambiguity. In the following, we consider proof-structures modulo neutrality of the constant 1 and associativity of connective ⅋.

Definition 4 (Small step reduction rule). *Let $\rightarrow$ be the reduction relation given by: if $\forall v$ a literal of ψ, $v \not\rightsquigarrow^* x^{\perp}$ then*



Theorem 1 ((small steps) Correctness criterion). *Let M be a closed BM, M is correct iff $M^o \rightarrow^* 1$.*

Briefly speaking, one can prove that the relation $\longrightarrow$ and the inverse relation are stable wrt DR-correctness by induction over the height of ψ. One may want to get rid of the (global) condition in favor of a local condition. This is possible thanks to the structure of modules. Suppose M is a correct closed module, then one may define an equivalent proof-net by sufficiently adding fresh variables as described in the introduction. It is easy to prove that the constraint is satisfied by x or $x^{\perp}$ for each variable x.

[6] The type is sufficient to build a proof structure as by construction of modules axioms are uniquely defined. We abusively note $s(M)$ in place of $s(M^o)$ in the following.

However, the reduction system being not strongly confluent, a reduction on a variable may lead to a proof structure on which the condition is not always satisfied. There are two cases where this does not happen: either all variables on a tensor have their negation on the same ⅋, or the converse interchanging ⅋ and $\otimes$. The (big step) reduction relation $\twoheadrightarrow$ in Fig. 1 uses this fact. Note that this system is confluent and terminates.

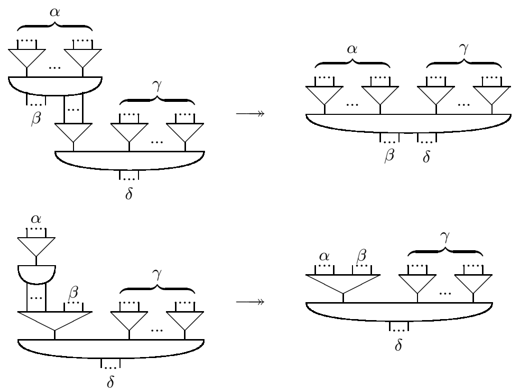
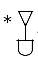


Fig. 1. Big step reduction relation.

Proposition 2 (Stability). *Let M and N be two closed modules and $M \twoheadrightarrow N$, M is c-correct iff N is c-correct.*

Proof. One can define a function from left switched module onto right switched module stable wrt acyclicity, connectedness, and the inverse properties. □

Theorem 2. *A closed module M is c-correct iff $M \twoheadrightarrow^*$* .

Proof. As the reduction rules are stable wrt correctness, it remains to prove that a correct non-terminal closed module M can always be reduced. We define a partial relation on negative poles: a negative pole is smaller than another one if there exists a positive pole s.t. the first negative pole is linked to the bottom of the positive pole and the second negative pole is linked to the top of the positive pole. We consider the transitive closure of this relation.

If maximal negative poles do not exist then there exists at least one cycle in the module alternating positive and negative poles. We can then define a switching function on the module (choosing the correct links for negative poles) s.t. the switched module has a cycle. Hence contradiction.

So let us consider one of the maximal negative pole, and the corresponding positive pole. We remark that such a negative pole has no outcoming links (the module is closed and the negative pole is maximal). If the positive pole has other negative poles, we can omit the maximal negative pole by neutrality. Otherwise, let us study the incoming negative poles.

If there is no such incoming link, then M is the terminal module. If each incoming negative pole has at least one link going to another positive pole, then one can define a switching function using for each of these negative poles one of the link that does not go to the positive pole we considered first. Hence the switched module is not connected (there are no outgoing links). Hence contradiction. So there exists at least one incoming negative pole with the whole set of links associated to the positive pole: the first rule applies and we are finished. □

Note that this proof extensively uses the bipolar nature of modules. Moreover, the proof may have been given considering minimal poles in place of maximal poles, and for each proof only one of the two reduction rules is sufficient and necessary! Finally, the same technique as Guerrini [10] used for Danos criterion may be applied here to get a linear algorithm. The technique we present here is quite close to the one used by Bechet [5]. However his definitions of modules were more restrictive, and the application concerne mainly non commutative logic.

4 Open modules

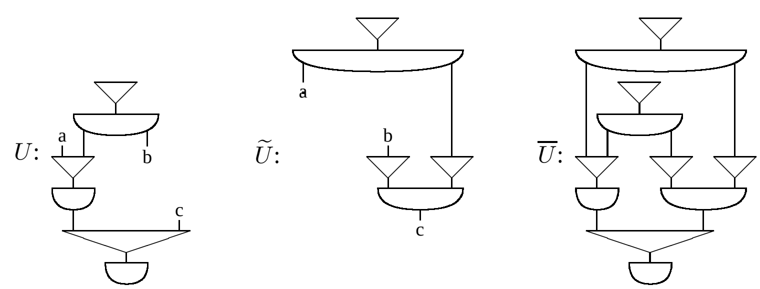
We focus in this section on open modules, i.e. partial polarized proof-structures. An *open module* is a possibly non closed BM. Studying correctness of open modules is a necessary step towards the specification of a logic programming language based on bipolar modules. We search for correctness criteria valid in the general case, hence extending Andreoli's works based on Murawski and Ong criterion. The criterion we give for closed modules is a good basis as it is well suited for bipolar modules and takes care simultaneously of acyclicity and connectedness.

4.1 O-correction

The bigstep reduction relation presented in the previous section is not sufficient to characterize again correctness of open module. Let U be the first module of the next example. Bigstep reductions on U leave the negative pole with variable a unchanged, hence the normal form is not the one required by the c-correctness theorem, though U has to be considered correct. The c-correctness theorem 2 cannot be straightfully extended to open modules.

Correctness of open modules is defined wrt correctness of closed extensions. We define a closed module N to be a *closure* of an open module M iff M is a submodule of N. Such closures are abusively noted $\overline{M}$ without referring to N when there is no ambiguity. As a BM is a graph with pending edges, one defines submodules and induced modules as expected. We use the notation $\widetilde{M}$ for the module $\overline{M}$ without M but with border $b(M)$ (cf def.2).

Example 4. In the next figure, $\overline{U}$ is a closure of U.



Note that he composition of U with a set of only initial/final EBMs is a closure too.

An open module M is *o-correct* iff there exists a c-correct closure of M. The open module U of the example is o-correct because the given closure is c-correct. Note that there is no other c-correct closure. Hence it is not possible in general to split the problem of finding a closure into finding a completion by initial modules and final modules. In the previous section, we defined a rewriting system able to test the correctness of a closed module. As this system is stable wrt connectedness and acyclicity, it is invariant wrt the Danos-Regnier criterion. In order to take care of open modules, we extend connectedness to connectability (acyclicity is treated easily) and prove that connectability and acyclicity are necessary and sufficient for o-correctness. However, we are not able to define a single rewriting system that commutes with composition. An open module M is *acyclic* if for all generalized switches s on M, $s(M)$ is acyclic. Note that a submodule of an acyclic module is obviously acyclic.

An open module M is *connectable* iff there exists a connected closure $\overline{M}$ s.t. $\widetilde{M}$ is acyclic. As a connected closed module is already connectable (just take itself as closure), the connectability is an extension of the connectedness property. We give an equivalent definition: an open module M is connectable iff the closed module $M \circ F$ is connected where F is a *full connector* EBM for M, i.e. F has as hypotheses the set of conclusions of M, is final if M has no hypothesis or has a negative pole with one conclusion for each of its hypotheses. In fact if there exists a connected closure $\overline{M}$ then $M \circ \widetilde{M}$ is connected. So *a fortiori*, $M \circ F$ is connected. The converse comes from the definition.

Theorem 3 (o-correctness). *An open module M is o-correct iff M is acyclic and connectable.*

Proof. By definition o-correctness implies acyclicity and connectability. If M is acyclic and there exists a connected closure $\overline{M}$ st $\widetilde{M}$ is acyclic then by induction on the number of cycles of $\overline{M}$, one can construct an acyclic and connected closure of M.
If there is a cycle σ in $\overline{M}$ then by hypothesis $\sigma \cap b(M) \neq \emptyset$. Suppose there exists a hypothesis of M $h \in \sigma \cap b(M)$, one defines N to be $\widetilde{M}$ where we substitute a fresh label h' to h. Let N' be the composition of the initial EBM of border $\{h\}$, the final EBM of border $\{h'\}$ and N. $M \circ N'$ has one cycle less than $\overline{M}$ and is a connected closure.
Otherwise the elements of $\sigma \cap b(M)$ are conclusions of M. Let c be such a conclusion. We consider the following cases:

- if c in $\sigma \cap b(M)$ is the only conclusion of a negative pole n, then one can do the same thing as in the previous case.
- else let d be a conclusion in $\sigma \cap b(M)$ distinct from c of n. One renames c (resp. d) in $\widetilde{M}$ in c' (resp. d') to get N. One defines also an EBM D with one conclusion d' and two hypotheses c and d, and an initial EBM E with conclusion c'. Then $X = M \circ D \circ E \circ N$ is a connected closure of M and $D \circ E \circ N$ is acyclic. Hence X is a connected closure of $M \circ D$ and $E \circ N$ is acyclic. We suppressed the cycle σ. However, it may be the case that there were a cycle through d and D doubles it ! For that purpose, we transform M to get rid of this extra cycle. Let M' be M where

we identify the two edges labelled c and d in one labelled d'. Then $M' \circ E \circ N$ is a connected closure of M' and $E \circ N$ is acyclic. Moreover the number of cycles in $M' \circ E \circ N$ is one less than in $\overline{M}$. Thus there exists N' acyclic such that $M' \circ N'$ is c-correct. Hence $M \circ D \circ N'$ is c-correct. $\square$

4.2 Acyclicity criterion: a contraction relation $\twoheadrightarrow$

An open module M *restricted to the subset* I of $b(M)$ is the subgraph of M where we omit pending edges not in I. We denote it $M\lfloor_I$. Informally an open module M restricted to I is a submodule of border I. The restriction of an open module to the empty set is a closed module. Restriction gives naturally an equivalent definition of acyclicity for open modules: an open module M is *acyclic* iff the closed module $M\lfloor_\emptyset$ is acyclic. Hence the proposition given in the previous section applies:

Proposition 3 (acyclicity). *An open module M is acyclic if $M\lfloor_\emptyset \twoheadrightarrow^* \overset{\triangledown}{\sqcup}$.*

Proof. $M\lfloor_\emptyset$ is a closed module and $M\lfloor_\emptyset \twoheadrightarrow^* \overset{\triangledown}{\sqcup}$ then by stability of acyclicity (of the inverse relation) $M\lfloor_\emptyset$ is acyclic. M is then acyclic. $\square$

Note that the converse is not true, otherwise acyclic closed modules would be correct! A way to characterize acyclicity by means of a reduction relation is to enlarge the reduction $\twoheadrightarrow$ (quotienting the set of normal forms). Splitting the negative poles suffices to continue reduction until we get a non-empty set of $\overset{\triangledown}{\sqcup}$: closing modules may link disjoint connected components. It is then obvious to deduce a necessary and sufficient condition for acyclicity. Andreoli considered in [4] only transitory proof-structures. A *transitory proof-structure* is equivalent to a BM without hypothesis[7] such that negative poles have always conclusions and obtained by a bottom-up composition of EBMs. As negative poles have pending edges, there is always a way to connect it to other parts of the module: if a transitory module M is acyclic then M is connectable. Hence a transitory module M is o-correct iff M is acyclic. The reduction relation we give to test acyclicity can be considered as an alternative to Andreoli's method.

4.3 Connectability criterion: a contraction relation $\longrightarrow_c$

The proof of the correctness of the big step reduction relation for closed modules gives the keys for finding a connectability property that relies on the structure of an open module (and not on the modules candidate to close it !). Proof of theorem 2 is based on reducing first maximal negative poles. In the case of open modules, maximal elements may have pending edges that should be connected in the closure. But we notice that we keep connectability if we replace the whole set of pending edges for such an element by just one pending edge. With this in mind, we consider the (non directed) contraction relation of Fig. 2 on (contracted) modules. The first three rules are a n-ary formulation of Danos contraction relation. Danos [6] proved correctness of the relation for (closed) proof-structures only, though we extend the results to (open) bipolar modules.

---

[7] In fact, there may be hypotheses in built modules but these are unused.

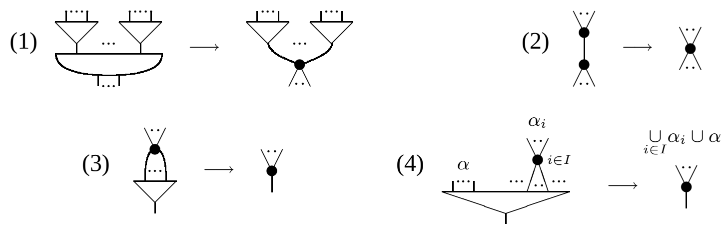
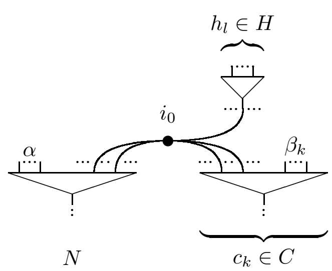


Fig. 2. Contraction relation.

Rule (4) is restricted to cases where the negative pole is such that for all $i \in I$, $\alpha_i \cap b(M) \neq \emptyset$ and $\alpha \subseteq b(M)$ where $b(M)$ is the set of pending edges of M, i.e. the border set. The sets I and α may be empty. We denote by $\rightarrow_c$ one rewriting step and by $\rightarrow_c^*$ the reflexive and transitive closure of $\rightarrow_c$. We call *contracted node* a black node. Note that rule (4) is simply the rewriting of a negative pole in a contracted node if the condition is satisfied. Thus acyclicity is not preserved but connectability is.

Proposition 4. *The relation $\rightarrow_c^*$ is strongly confluent and terminates.*

Proof. The first rule acts just as a mark. We can forget it: it is just for convenience. Each rule applies locally and strictly decreases the number of negative poles and contracted nodes. The rules are disjoint except for a pair of negative poles linked by the same contracted node i_0 for which rule (4) can be applied (it is a trivial case), and except in the particular case where the left hand side of rule 4 is reduced to the one of rule 3: in this case the results are identical. □

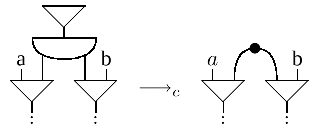


We extend the notions of switching to modules with contracted nodes: contracted nodes are treated as positive poles. Acyclicity, connectedness, closure and connectability are extended in the same way. As in section 3, our strategy consists of characterizing amongst normal forms of this relation the correct ones, and prove stability of, say, connectability.

Let M be an open module and f the corresponding normal form. By definition if f does not contain a negative pole then f is a set of contracted nodes $\{n_j\}_{j\in J}$ s.t. all pending egdes are in $b(M)$. We use the notation **cc** for a set of contracted nodes $\{n_j\}_{j\in J}$ s.t. for all $j \in J$ n_j has at least one edge in the border $b(M)$ except if $\mid J \mid = 1$. If f contains a negative pole N then, f being a normal form of relation $\rightarrow_c$, rule (4) does not apply on N. Hence the set I as defined by rule (4) is st there exists $i_0 \in I$, $\alpha_{i_0} \cap b(M) = \emptyset$. Moreover this contracted node i_0 is linked to hypotheses of negative poles $\{h_l\}_{l\in L}$ and to conclusions of only negative poles $\{c_k\}_{k\in K}$ st each of them has other conclusions $\beta_k \neq \emptyset$ not linked to i_0 (otherwise rule (2) applies for such nodes): see figure just above.

If we suppose the negative pole N is a maximal one (i.e. $H = \emptyset$), there is a switching (on α or on some $i \neq i_0$ and on one of each β_k) s.t. f (as closures of f) is not connected. Thus f is not connectable.

Example 5. The following subform implies not connectability:



Proposition 5 (stability). *Connectability is stable wrt (resp. inverse) contraction rules.*

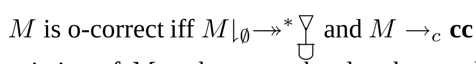
Proof. The three first rules satisfy obviously stability as does the reverse relation. Let M be an open module s.t. $M \rightarrow_c M'$ by the contraction rule (4) and there exists $\overline{M}$ connected and $\widetilde{M}$ acyclic. Obviously $M' \circ \widetilde{M}$ is connected. Concerning stability of the inverse relation, let M be an open module s.t. $M \rightarrow_c M'$ by the contraction rule (4) and let F be a full connector EBM for M'. Note that $b(M') = b(M)$. The connectability of M' implies that $M' \circ F$ is connected. Wrt rule (4), because for all $i \in I$, $\alpha_i \cap b(M) \neq \emptyset$ and $\alpha \subseteq b(M)$, for every switches s, $s(M \circ F)$ is connected too. □

By stability of connectability of the relation and its inverse we have:

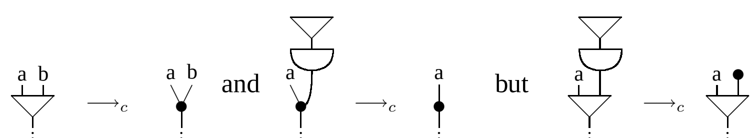
Theorem 4. *Let M be an open module, M is connectable iff $M \rightarrow_c^*$ **cc**. Hence an open module M is o-correct iff M is acyclic and $M \rightarrow_c^*$ **cc**.*

5 Composition of modules

In the sequel we discuss an incremental criterion to test the composition of an open module with an EBM. Let M be an o-correct open module and E an EBM s.t. $b(M) \cap b(E) \neq \emptyset$ (otherwise the test is easy). As seen above, acyclicity and connectability, hence o-correctness, of M may be decided by computing normal forms. Our aim is to decide the o-correctness of the composition $M \circ E$ 'incrementally' i.e. not directly but o-correctness of M being given. This leads us to define a specific contraction relation $\rightarrow_w$ to replace $\rightarrow_c$. From the previous section we have:



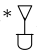
Because of the restriction of M to the empty border, the acyclicity condition given above does not commute with composition. It is the same for connectability: even if there is preservation of the border with $\rightarrow_c$, a choice is made for the completion of M which may be different from the way composition with E is done. For example:

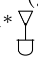
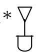


In the sequel we show that if we release the restriction operation we can incrementally manage acyclicity. The relax of the (implicit) completion in the rewriting rules dealing with connectability gives also an incremental criterion for connectability.

5.1 Incremental acyclicity: $\twoheadrightarrow$

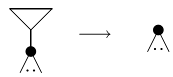
The restriction to empty set is stable wrt the reduction $\twoheadrightarrow$ i.e. if M is an open module s.t. $M \twoheadrightarrow N$ then $M\lfloor_\emptyset \twoheadrightarrow N\lfloor_\emptyset$. Hence an incremental test for acyclicity follows:

Proposition 6. *Let M be an open module s.t. $M \twoheadrightarrow^* f$ and E an EBM. $M \circ E$ is acyclic if $(f \circ E)\lfloor_\emptyset \twoheadrightarrow^*$* .

Proof. If $M \twoheadrightarrow^* f$ then $(M \circ E) \twoheadrightarrow^* (f \circ E)$. Following previous remark, $(M \circ E)\lfloor_\emptyset \twoheadrightarrow^* (f \circ E)\lfloor_\emptyset$. Thus if $(f \circ E)\lfloor_\emptyset \twoheadrightarrow^*$  then $(M \circ E)\lfloor_\emptyset \twoheadrightarrow^*$ . □

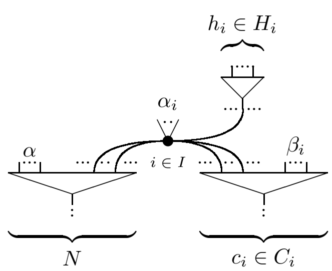
5.2 Contraction relation (without completion): $\rightarrow_w$

We consider the rewriting system given to test connectability where rule (4) is restricted to the following degenerated case ($\alpha = I = \emptyset$ and application of rule (2)):



We denote by $\rightarrow_w$ one rewriting step and by $\rightarrow_w^*$ the reflexive and transitive closure of $\rightarrow_w$. As it is a subsystem of the previous one, the relation $\rightarrow_w^*$ terminates and is still strongly confluent (there is only trivial independant pairs).

We study the normal forms. By definition an open module contracts in a normal form composed with only contracted nodes or contracted modules where each negative pole N is of the following form:



- I is a (possibly empty) set of contracted nodes,
- each $i \in I$ is linked to a set C_i of other negative poles by conclusions and to a set H_i of other negative poles by hypothesis (the sets C_i and H_i may be empty). Moreover for all $c_i \in C_i$ $\beta_i \neq \emptyset$,
- α and α_i are (possibly empty) subsets of $b(M)$ for all $i \in I$.

We focus on the two possible forms of negative pole:

- there exists $i_0 \in I$ s.t. $\alpha_{i_0} = H_{i_0} = \emptyset$. We denote such forms by **notcc**.
- for all $i \in I$, $\alpha_i \neq \emptyset$ or $H_i \neq \emptyset$. These negative poles may be considered in the previous system $\rightarrow_c^*$.

If a normal form has no negative poles then it is a set of contracted nodes. We add to the **notcc** forms the case where there is at least one contracted node without pending edges and other nodes.

In order to compare these normal forms with the normal forms of $\rightarrow_c$ observe that: (i) by definition of normal forms, if $I = \emptyset$ then $\alpha \neq \emptyset$, and if $I \neq \emptyset$ then $\mid I \mid \geqslant 2$ or $\alpha \neq \emptyset$, (ii) for all $i \in I$ for all $c_i \in C_i$ we have $\beta_i \neq \emptyset$. It follows that if a normal form

g wrt $\rightarrow_w^*$ of an open module M contains a **notcc** subform then there is a generalized switch s.t. g is not connected. The stability of connectedness wrt $\rightarrow_w^*$ being given, M is not connected (neither its closures), thus not connectable.

Remark that the **notcc** forms are already in the previous system: they are normal forms which are not the **cc** forms! In fact the **notcc** subforms are invariant wrt the previous system $\rightarrow_c^*$. Moreover as stability of connectability of the inverse relation is easily proven, we have:

Theorem 5. *Let M be an open module, M is connectable iff $M \rightarrow_w^* g$ s.t. **notcc** $\notin g$.*

Proof. Let M be s.t. $M \rightarrow_w^* g$. If **notcc** $\in g$ then g is not connected (neither its closures) and by stability of connectedness M is not connectable. Conversely, if **notcc** $\notin g$ then $g \rightarrow_c^*$ **cc** by invariance of **notcc** wrt $\rightarrow_c^*$. By theorem 4, g is connectable. The result is obtained by stability of connectability of the inverse relation wrt $\rightarrow_w^*$. □

Hence, an open module M is o-correct iff M is acyclic and $M \rightarrow_w^* g$ s.t. **notcc** $\notin g$. By confluence property and theorem 5 we have an incremental test: Let M be a connectable open module s.t. $M \rightarrow_w^* g$ and E an EBM s.t. $b(M) \cap b(E) \neq \emptyset$. We have:

$$M \circ E \text{ is connectable iff } f \circ E \rightarrow_w^* g \text{ s.t. } \textbf{notcc} \notin g.$$

5.3 A test for composition

Testing the composition of an EBM E on a correct module M may be done in the following way. We associate to such a module M a pair (f, g) such that $M \twoheadrightarrow^* f$ and $M \rightarrow_w^* g$. We compute the pair (f', g') associated to $M \circ E$: $f \circ E \twoheadrightarrow^* f'$ and $g \circ E \rightarrow_w^* g'$. Then E may be plugged onto M, i.e. the composition is correct, iff $f' \lfloor_\emptyset \twoheadrightarrow^* \begin{smallmatrix}\bigtriangledown \\ | \\ \sqcup\end{smallmatrix}$ and **notcc** $\notin g'$. This test may be implemented in such a way that pre-computations are done in M in order to optimize the test. Moreover this allows for a concurrent treatment for testing composition by only locking a reduced part of the module M.

6 Conclusion

Studying the correctness of open modules is a necessary condition towards incremental composition of partial proof-nets. Furthermore their concurrent construction allows for a new approach in designing logic programming languages besides standard ones [1, 11, 13]. In the Horn fragment as well as with linear logic, 'classical' logic programming is based on a step by step reduction of goals to be proven by means of a resolution or a progression rule, i.e. the correctness of a computation is reduced to a pattern recognition between some part of the current goal and the head of a chosen clause. More complex than the propositional Horn fragment, pattern recognition is done wrt the whole current environment when considering, e.g. the full linear logic [12]. In all these cases, the operational model is unable to reveal possible concurrent computations. A contrario, the proof net approach is a natural framework as each proof net represents a whole bunch of sequentialized computations: commuting rules lead to the same proof net.

For that purpose, we first extend the classical rewriting criterion of Danos to the n-ary bipolar case for testing the correctness of closed modules. We show in particular that polarization greatly simplifies the rewriting procedure. We finally modify the criterion to take care of open modules proving that correctness of open modules reduces to testing linearly acyclicity and connectability. This includes Danos results in a more general framework. It also extends Andreoli's works by removing constraints on objects we consider.

An interesting remaining question is to take care of exponential modalities in polarized and partial proof-structures. Even if Andreoli proves the focalisazion property for the whole linear logic, management of exponentials with proof nets requires extra structure such as boxes, i.e. bounded regions, as their behaviour is context dependent. In our opinion, this could yield a local management of the boxes, just considering transformation on part of the region border thanks to polarization and focalisation. Our characterization of proofs as a composition of complex objects can then be extended to multiplicative exponential polarized proof-structures in the same spirit, i.e. by (concurrently) reducing such structures.

References

1. Jean-Marc Andreoli. Logic programming with focusing proofs in linear logic. *Journal of Logic and Computation*, 2(3):297–347, 1992.
2. Jean-Marc Andreoli. Focussing and proof construction. *Annals of Pure and Applied Logic*, 107(1–3):131–163, 2001.
3. Jean-Marc Andreoli. Focussing proof-net construction as a middleware paradigm. In A. Voronkov, editor, *CADE*, volume 2392 of *Lecture Notes in Computer Science*, pages 501–516. Springer, 2002.
4. Jean-Marc Andreoli and Laurent Mazaré. Concurrent construction of proof-nets. In M. Baaz and J. A. Makowsky, editors, *CSL*, volume 2803 of *Lecture Notes in Computer Science*, pages 29–42. Springer, 2003.
5. Denis Béchet. Incremental parsing of lambek calculus using proof-net interfaces. In ACL/SIGPARSE, editor, *Eigth International Workshop on Parsing Technologies*, Nancy, France, April. citeseer.ist.psu.edu/668958.html.
6. Vincent Danos. *Une application de la logique linéaire à l'étude des processus de normalisation (principalement de λ-calcul)*. PhD thesis, Université Denis Diderot, Paris 7, 1990.
7. Vincent Danos and Laurent Regnier. The structure of multiplicatives. *Archive for Mathematical Logic*, 28(3):181–203, 1989.
8. Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.
9. Jean-Yves Girard. On the unity of logic. *Ann. Pure Appl. Logic*, 59(3):201–217, 1993.
10. Stefano Guerrini. Correctness of multiplicative proof nets is linear. In *LICS*, 1999.
11. Dale Miller. Forum: A multiple-conclusion specification logic. *Theor. Comput. Sci.*, 165(1):201–232, 1996.
12. Dale Miller. Overview of linear logic programming. In T. Ehrhard, J.-Y. Girard, P. Ruet, and P. Scot, editors, *Linear Logic in Computer Science*, volume 316 of *London Mathematical Society Lecture Note*. Cambridge University Press, 2004.
13. Dale Miller, Gopalan Nadathur, Frank Pfenning, and Andre Scedrov. Uniform proofs as a foundation for logic programming. *Ann. Pure Appl. Logic*, 51(1-2):125–157, 1991.
14. Andrzej Murawski and Luke Ong. Dominator trees and fast verification of proof nets. In *LICS'00*, pages 181–191. IEEE Computer Society Press, 2000.