Math. Struct. in Comp. Science (1993), *vol.* 11, *pp.* 1–000

Minimality of the Correctness Criterion for Multiplicative Proof Nets

DENIS BECHET

CRIN-CNRS & INRIA–Lorraine
Boîte Postale 239
54506 Vandœuvre-les-Nancy, France
e-mail: Denis.Bechet@loria.fr

Received

Almost a decade ago, Girard invented linear logic with the notion of proof-net. Proof-nets are special graphs built from *formulas*, *links* and *boxes*. However, not all nets are proof-nets. Firstly, they must be well constructed (we say that such graphs are proof-structures). Secondly, a proof-net is a proof-structure that corresponds to a sequential proof. It must verify a correctness criterion. One may wander what this static criterion means for cut-elimination. We prove that every not-correct proof-structure (without cut) can be put in an environment where reductions lead to two kinds of wrong basically configurations: dead-locks and disconnected proof-structures. Thus, this proof says that there does not exist a bigger class than proof-nets of proof-structures where normalization does not lead to very bad configurations.

1. Introduction

There exists two formalisms for presenting a proof in linear logic (Girard, 1987). The usual one uses sequents and rules. The second one used special graphs known as *proof-nets*. From a syntactical point of view, a well-formed proof tree in sequent calculus always gives a proof. To check that we really have a proof, we only need to look at the local derivations and see if they are instances of the inference rules. This property is no more valid in the framework of proof-nets. A *proof-structure* may be well-formed but it may not correspond to a proof. We need a global criterion (Danos and Regnier, 1989) which is not syntactical to distinguish between proof-structures that correspond to a sequential proof to the ones that correspond to nothing.

Here, we are interested by not correct proof-structures which do not correspond to a sequential proof, trying to show why they are bad. In fact, it is not obvious to reject bad proof-structures simply because they are not sequentializable. They are perhaps well funded. For instance, we can add the Mix rule (Fleury and Retoré, 1994) to *plain linear logic*. Thus, the criterion that distinguishes proof-nets from other proof-structures is looser. More proof-structures become proof-nets. In fact, we need a justification for the rejection of bad proof-structures that is different to the translation of sequential proof into

proof-nets. Here, this new justification is based on *wrong configurations*, cut-elimination and instantiation of atomic formulas.

Firstly *basically wrong proof-structures* are proof-structures that contain a *dead-lock* or that are *disconnected*. Secondly, we extend the notion of wrongness to proof-structures that reduce to a basically wrong proof-structure. Those proof-structures are obviously not satisfactory and must be definitively rejected. The next step consists to plug proof-structures into environment (i.e. to connect the conclusions of a proof-structure to other proof-nets) and to see if we have created a wrong proof-structure. The last operation consists in instantiation of atomic formula.

In our point of view, a proof-structure that must be definitively rejected as a logical object is:

1 A proof-structure that contains a *basically wrong configuration*: a *dead-lock* which is a definitively frozen object or a *disconnected graph* which seems to be incorrect if we reject the Mix-rule.
2 A proof-structure that reduces to one of point 1. Here, cut-elimination is the crucial mechanism.
3 A proof-structure such that there exists an environment that creates a proof-structure of point 2. This point says that a good logical object must interact with every good environment.
4 A proof-structure such that there exists an instantiation of its atomic formula that leads to a proof-structure of point 3. This means that a proof is also a proof for any substitution of its atomic formula. In other words, a proof of a sequent is implicitly universally quantified.

The main theorem of this paper concludes that for multiplicative linear logic, proof-structures that are not correct (i.e. with respect to the criterion) and that do not contain any cut are bad with respect to point 4. In other words, this proof means that it is not possible to find a bigger class of proof-structures without cut than the proof-nets that is consistent with the 4 points: no basically wrong configuration, stable by cut elimination, modularity and instantiation.

2. Multiplicative linear logic

2.1. *Sequential Proof*

This section gives some definitions used here upon linear logic, proof-structures and proof-nets. The fragment studied is plain multiplicative linear logic without constant. Formulas are inductively constructed from atomic formulas $X, X \in \mathcal{V}$ and their dual $X^{\perp}, X \in \mathcal{V}$ where $\mathcal{V}$ is a set of propositional variables and from the two binary connectives $\otimes$ and $\parr$. Formulas are noted by the symbols A, B,... The formulas follow the usual laws of De Morgan: $(A \otimes B)^{\perp} \stackrel{\mathbf{Def}}{=} A^{\perp} \parr B^{\perp}$, $(A \parr B)^{\perp} \stackrel{\mathbf{Def}}{=} A^{\perp} \otimes B^{\perp}$ and $(A^{\perp})^{\perp} \stackrel{\mathbf{Def}}{=} A$. Sequents are multi-sets of formulas and are written $\vdash A_1, \cdots, A_n$. The symbols Δ, Γ denote a multi-set of formulas. The rules are given in figure 1.

Definition 1. A *sequential proof* is a proof in this system.

$$\frac{}{\vdash A, A^{\perp}}\ \mathbf{Ax} \qquad \frac{\vdash \Gamma, A \quad \vdash A^{\perp}, \Delta}{\vdash \Gamma, \Delta}\ \mathbf{Cut}$$

$$\frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \parr B}\ \parr \qquad \frac{\vdash \Gamma, A \quad \vdash B, \Delta}{\vdash \Gamma, A \otimes B, \Delta}\ \otimes$$

Fig. 1. Sequential Rules

2.2. *Proof-Nets*

On the second hand, we have proof-structures, proof-nets and a criterion for distinguishing proof-nets from other proof-structures.

Definition 2. A *link* is one of the 4 expressions listed in figure 2.
Formulas above the line are the *premises* and formulas under the line are the *conclusions*.
A *module* is a multi-set of formulas linked together by different links such that a formula is premise of at most one link and is conclusion of at most one link.
A *proof-structure* is a module such that each formula is premise of exactly one link.
The *frontier* of a module (or a proof-structure) is the multi-set of formulas that are premise or conclusion of at most one link.

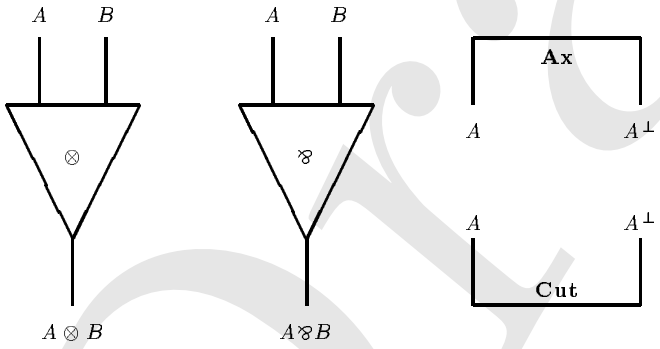


Fig. 2. Links

To a sequential proof, one may associate a proof-structure simply by forgetting the proof structure, retaining only the combinatorial compounds of the proof.

Definition 3. A *proof-net* is a proof-structure that corresponds to a sequential proof.

Here comes an important fact, which is that not all proof-structures are proof-nets. But, there exist criterions that tell us if a proof-structure is or is not a proof-net. In this article, we used the Danos-Regnier criterion (Danos and Regnier, 1989) which is divided in two sub-criterions.

Definition 4. A switching of a proof-structure (or a module) is a selection for every $\parr$-link between the *left* or the *right* position. This switching induces a graph by replacing each link by the vertices shown on figure 3.

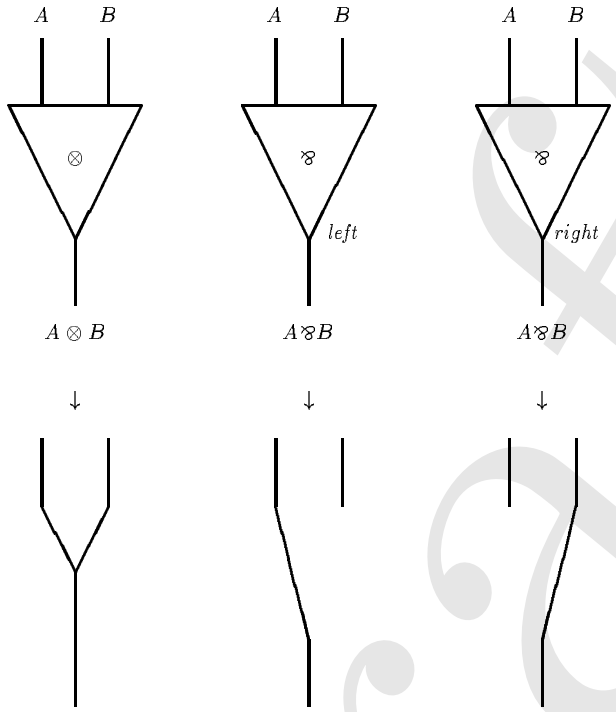


Fig. 3. Switching links

Definition 5. A proof-structure is correct (it verifies the correctness criterion) if and only if for every switching, the graph is:

— acyclic
— connected.

Theorem 6. Correct proof-structures are proof-nets (they correspond to a sequential proof).

This is the usual sequentialisation theorem on proof-nets. See (Girard, 1987; Danos, 1990).

2.3. *Cut elimination*

One can always eliminate cuts in a sequential proof. For proof-structures, we can define a cut-elimination mechanism as a graph rewriting system. The rules are given on figure 4. They eliminate or identify formulas and change links.

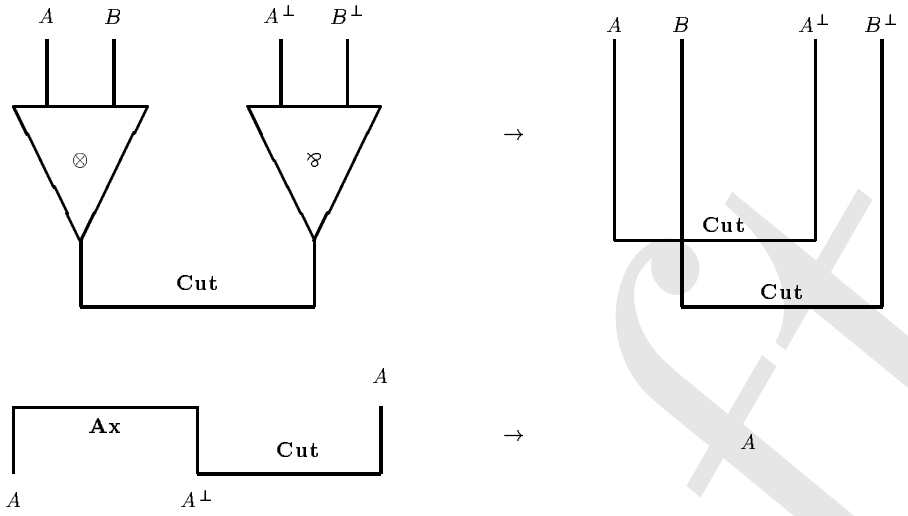


Fig. 4. Cut elimination rules

3. Wrong and bad proof-structures

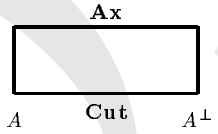
The correspondence between proof-nets and correct proof-structures says that proof-structures that are not correct do not correspond to a sequential proof. But are they really incoherent from a logical point of view? In the following, we will reject progressively proof-structures using different arguments.

3.1. *Wrong configurations*

The first step here is to reject proof-structures that contain basically wrong configurations. For plain multiplicative linear logic (the system studied here), there are two cases.

Definition 7. A *basically wrong configuration* is:

— a cut under an axiom called a *dead-lock*:



— a disconnected net.

A cut under an axiom is called a *dead-lock* because such a loop can never disappear or interact with the outside world. The condition on connectivity says that each conclusion of a proof-net can interact with all the others. This second configuration is acceptable for linear logic with Mix rule but not for plain linear logic.

Proof-structures that contain such basically wrong configurations are said to be *basically wrong proof-structures* and must not be considered as having a logical meaning.

The second argument extends this judgment to proof-structures that reduce to a basically wrong proof-structure. The argument lays on the fact that cut-elimination is one of the main principle of a logical system.

Definition 8. A proof-structure is wrong if and only if it reduces (in 0, 1 or more steps) to a proof-structure that contains a basically wrong configuration.

3.2. *Bad proof-structures*

The next argument for rejecting proof-structures comes from the use of environment. The idea here says that if a proof-structure has a logical meaning, it can be connected to other proof-nets such that the result has a logical meaning.

Definition 9. A proof-structure with n conclusions $A_1, \cdots, A_n$ is a *basically bad proof-structure* if and only if it exists n proof-nets $\mathcal{P}_1, \cdots, \mathcal{P}_n$ of n sequents $\vdash A_1^\perp, \Gamma_1$,...,$\vdash A_n^\perp, \Gamma_n$ such that the proof-structure obtained by connecting A_1 and $A_1^\perp$ ... A_n and $A_n^\perp$ by n cuts, is a wrong proof-structure (see figure 5).

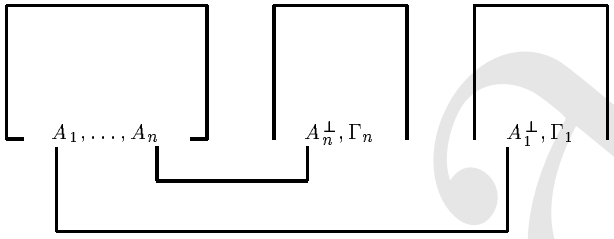


Fig. 5. Connection to an environment

Finally the last criterion extend rejection if it is possible to find an instantiation of the atomic formulas of a proof-structure such that the result is a basically bad proof-structure. The argument lays on the fact that substituing everywhere the atomic formulas of a logical object must give a logical object. It means that in a logical object, the free type variables are universally quantified.

Definition 10. A proof-structure is a *bad proof-structure* if and only if it exists a substitution of the atomic formulas such that the proof-structure obtained after the substitution is a basically bad proof-structure.

Remark:

The different classes are included:

— A basically wrong proof-structure is a wrong proof-structure (in 0 steps).
— A wrong proof-structure is a basically bad proof-structure (take as proof-nets, n axioms).
— A basically bad proof-structure is a bad proof-structure (take the identity substitution).

3.3. *Bad and correct proof-structures*

Of course, proof-net are not bad. They are proper logical objects.

Theorem 11. Proof-nets are not bad proof-structure.

Proof. This theorem lays on the fact that proof-nets are logical objects.

— Proof-nets do not contain a basically wrong configuration because a dead-lock contains a cycle (every switching has a cycle) and because proof-nets must be connected.
— A proof-net always reduces to a proof-net so a proof-net is not a wrong proof-structure.
— The connection of a proof-net into an environment gives a proof-net as it can easily be checked. So a proof-net is not a basically bad proof-structure.
— The instantiation of the atomic formulas of a proof-net gives a proof-net. A proof-net is not a bad proof-structure.

□

Remark:

The 4 classes are different. Figure 6 gives a wrong proof-structure which is not basically wrong. Figure 7 gives a basically bad proof-structure which is not wrong. The environment is a $\parr$ under an axiom. Figure 8 gives a bad proof-structure which is not a basically bad proof-structure. In fact, it is not possible to connect directly this proof-structure to a proof-net such that a dead-lock can appear. For that, we need, for instance, to instantiate B with $A^{\perp}$.

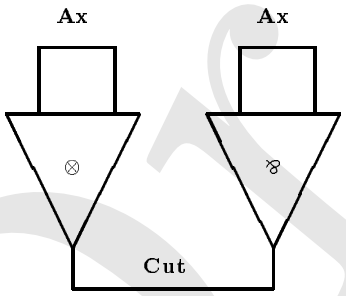


Fig. 6. Wrong but not basically wrong

4. Minimality of the correctness criterion

The main result of this paper consists on a proof of the minimality of the correctness criterion on proof-structures. This proof gives a justification for the choice of the class of proof-net (into proof-structure) as the class of the real logical objects and nothing else. This proof is mainly based on badness.

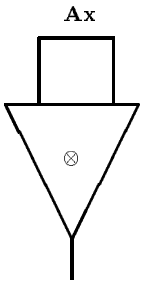


Fig. 7. Basically bad but not wrong

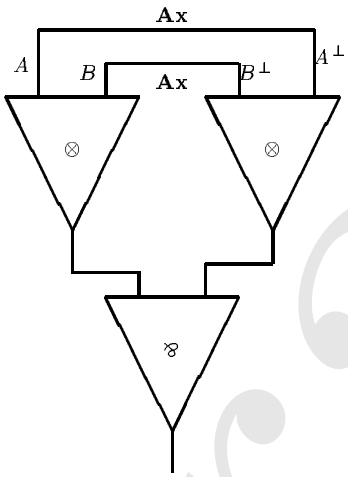


Fig. 8. Bad but not basically bad

Theorem 12. The bad proof-structures without cut are exactly the proof-structures that are not proof-nets (the proof-structures that do not satisfy the correctness criterion)

Remark:

This proposition works only for proof-structure without cut. In fact, it is well-known (see figure 9) that some not correct proof-structures may reduce to a proof-net. Those ones are not bad.

Proof. The proof is decomposed in several parts which are the lemmas below. □

Lemma 13. Let $\mathcal{N}$ be a proof-structure with n conclusions $A_1, \cdots, A_n$. $\mathcal{N}$ is bad if and only if the proof-structure with only one conclusion obtained from $\mathcal{N}$ by connected the n conclusions by $n-1$ $\parr$-links (in any order) is bad.

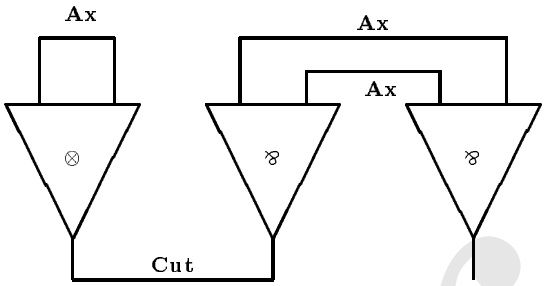


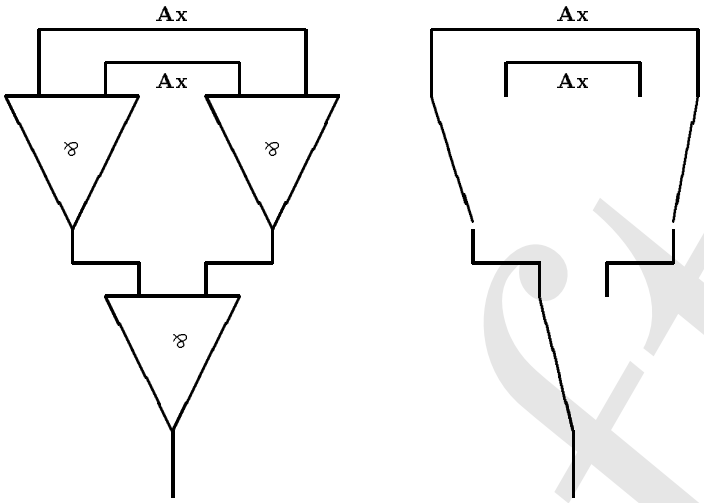
Fig. 9. Not correct but not bad

Proof. It is obvious when we see that the n proof-nets of the reveling environment for $\mathcal{N}$ creates a big proof-nets of the reveling environment for $\mathcal{N}$ with $n-1$ $\parr$-links when we connect them by $\otimes$-links and vice-versa. □

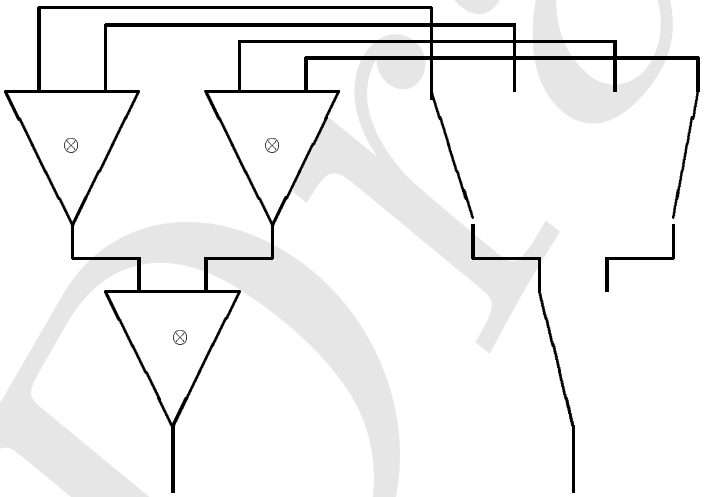
Lemma 14. Let $\mathcal{N}$ be a proof-structure without cut and with only one conclusion A. If there exists a switching of $\mathcal{N}$ such that the graph is not connected then we can find a proof-net with one conclusion of type $A^\perp$ such that the proof-structure obtained by connecting A and $A^\perp$ with a cut reduces to a disconnected proof-structure (a basically wrong proof-structure).

Proof. Let Σ be a switching of $\mathcal{N}$ such that the graph is not connected. The proof-structure $\mathcal{N}$ is composed of a tree $\mathcal{C}$ of $\otimes$-links and $\parr$-links and a set $\mathcal{A}$ of axiom links. Let $\mathcal{C}^\perp$ be the module obtained from $\mathcal{C}$ by substituting each $\otimes$-link by a $\parr$-link and each $\parr$-link by a $\otimes$-link. From $\mathcal{C}$ and $\mathcal{C}^\perp$, one can create a proof-net $\mathcal{I}$ with conclusions A and $A^\perp$ simply by connecting with axioms the premises of $\mathcal{C}$ to the premises of $\mathcal{C}^\perp$. Here, it is just a η-expansion of the axiom $\vdash A, A^\perp$. Now, using the switching Σ, we replace each left $\parr$ of the $\mathcal{C}$ side of the proof-net $\mathcal{I}$ by a new pending conclusion for the right premise and by identifying the left premise and the conclusion and we replace each right $\parr$ of the $\mathcal{C}$ side of the proof-structure $\mathcal{I}$ by a new pending conclusion for the left premise and by identifying the right premise and the conclusion. The new proof-structure $\mathcal{J}$ is, in fact, a proof-net. When this proof-net is connected to $\mathcal{N}$, after reduction of the cut which connects the module $\mathcal{A}$ to the remaining part of the $\mathcal{C}$ side of the proof-net $\mathcal{J}$, we obtain a proof-structure composed of only axioms and $\otimes$-links which is structurally identical to the disconnected graph given by the switching of $\mathcal{N}$.
This lemma uses the fact that in a proof-net (without cut), if we replace a $\parr$ by one of its switching position (see figure 3), we obtain a proof-net. □

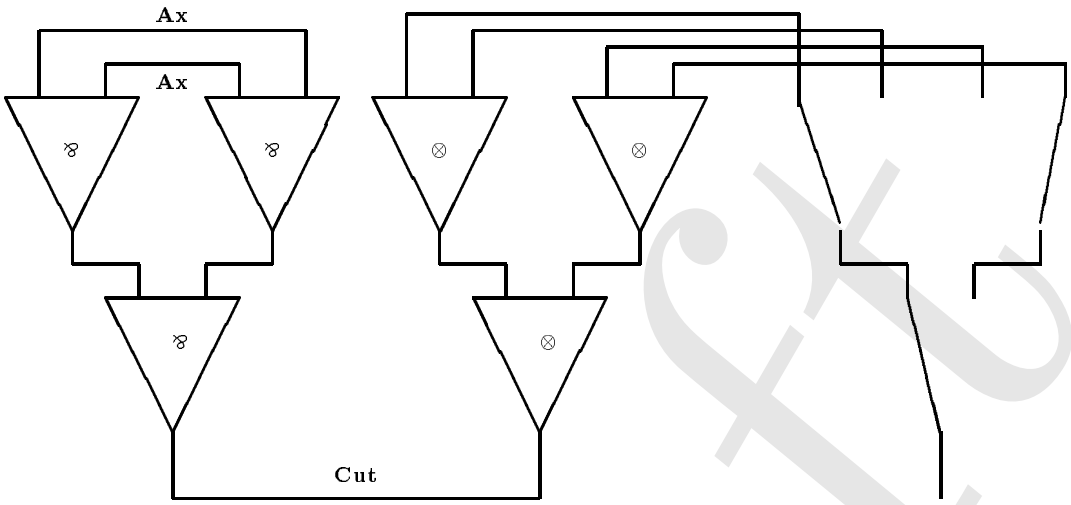
Here a small example of this construction is shown. The proof-structure below is not correct because for one of the switching, the graph is disconnected:



We transform the identity proof-net by replacing $\parr$-links on the left:



The connection of the bad proof-structure and the proof-net give a proof-structure that reduces to a disconnected proof-structure.



Lemma 15. Let $\mathcal{N}$ be a proof-structure without cut and with only one conclusion A. If there exists a switching of $\mathcal{N}$ such that the graph has a cycle then we can find a proof-net with one conclusion of type $\Phi(A^\perp)$ such that the proof-structure obtained by connecting $\Phi(A)$ of $\Phi(\mathcal{N})$ and $\Phi(A^\perp)$ with a cut reduces to a proof-structure containing a deadlock (a basically wrong proof-structure). Φ is the substitution of atomic formulas that instantiate $X, X \in \mathcal{V}$ by $(Z \otimes W)\parr(W^\perp \parr Z^\perp)$ and $X^\perp, X \in \mathcal{V}$ by $(Z^\perp \parr W^\perp) \otimes (W \otimes Z)$.

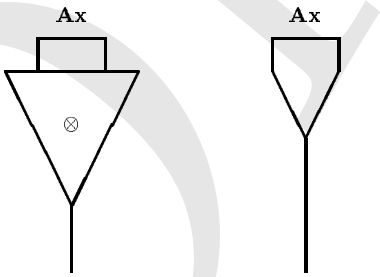
Remark:

The instantiation Φ is independent of the bad proof-structure.

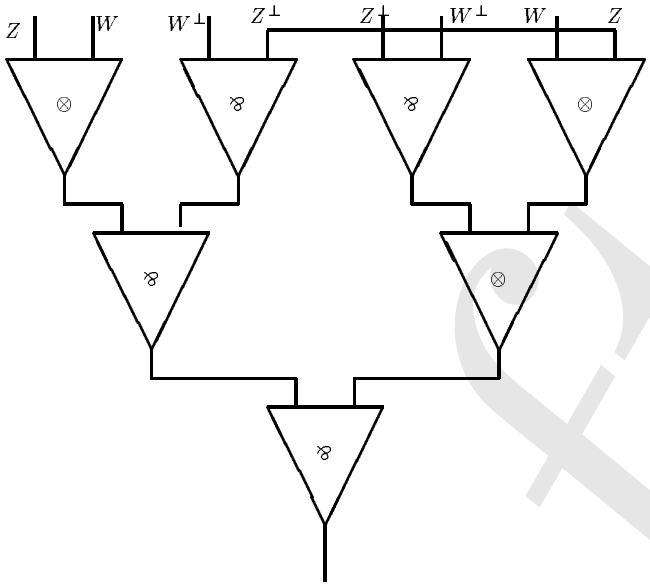
Proof. The switching of $\mathcal{N}$ creates a cycle in the graph. This cycle induces a cycle in the proof-structure. This cycle goes successively up and down. For instance, it starts on a conclusion A_1 of an axiom, it goes through this axiom, goes down the other conclusion $A_1^\perp$. It continues down through several links until it reaches one premise B_1 of a $\otimes$-link. Then it goes up through the second premise C_1 of the same $\otimes$-link. It goes up until it reaches the second axioms etc... Thus, the cycle crosses alternatively n axiom links $A_1/A_1^\perp, \cdots, A_n/A_n^\perp$ and n $\otimes$-links with premises $B_1/C_1, \cdots, B_n/C_n$.
Let $\mathcal{T}$ be the tree of $\otimes$-links and $\parr$-links corresponding to the formula $\Phi(A^\perp)$. For each formula A_i or $A_i^\perp$, it corresponds a sub-tree of $\mathcal{T}$ of type $\Phi(A_i)$ or $\Phi(A_i^\perp)$. The formulas $\Phi(A_i)$ and $\Phi(A_i^\perp)$ have a most left occurrence of one of the formulas $(Z \otimes W)\parr(W^\perp \parr Z^\perp)$ or $(Z^\perp \parr W^\perp) \otimes (W \otimes Z)$. For those occurences, there are exactly one occurrence of Z and one occurrence of $Z^\perp$. As a consequence, it is always possible to connect the Z occurrence corresponding to $A_i^\perp$ to the $Z^\perp$ occurrence corresponding to A_{i+1} (A_1 if $i = n$) with an axiom link.
The resulting module $\mathcal{M}$ is a *correct module* (see section 5 for the definition and theorems on correct modules). It may be completed by another correct module (see theorem 18) on its premises such that the result is a proof-net ($\Phi(A^\perp)$ is one of its conclusions).

Now, if we connect this proof-net to $\Phi(\mathcal{N})$, we obtain a proof-structure that reduces to a proof-structure with a dead-lock. This fact is already true with the module composed of $\Phi(\mathcal{N})$ and the constructed module $\mathcal{M}$. This module reduces first to a module where the n sub-trees $\Phi(A_1), \cdots, \Phi(A_n)$ are connected to $\Phi(A_1^\perp), \cdots, \Phi(A_n^\perp)$ by n cuts (which come from the proof-structure $\Phi(\mathcal{N})$) and with the n original axioms of $\mathcal{M}$. Then each couple reduces in such a way that the resulting module has an alternation of n cuts and n axioms which reduce to a dead-lock.
The last point concerns the fact that, from $\mathcal{M}$, we can find another module such that the union is a proof-net. This property is a general result on modules which is due to the fact that this module is correct. This module is correct because for every switching of $\mathcal{M}$, the graph induces has no cycle and has no internal component, not connected to its frontier (its premises and conclusions). There is no cycle because such a cycle must cross some axioms and some $\otimes$-links which is impossible by construction. In fact, a cycle in this module would induce a cycle in the proof-net of the identity $\vdash \Phi(A^\perp), \Phi(A)$ which is impossible. Finally there exists no internal component because the W and $W^\perp$ premises connect each component to the frontier of $\mathcal{M}$. Now, because $\mathcal{M}$ is a correct module and has only one conclusion and no cut, we can find another module and connect the premises of it to the premises of $\mathcal{M}$ to have a proof-net. This property is proved in the next section. □

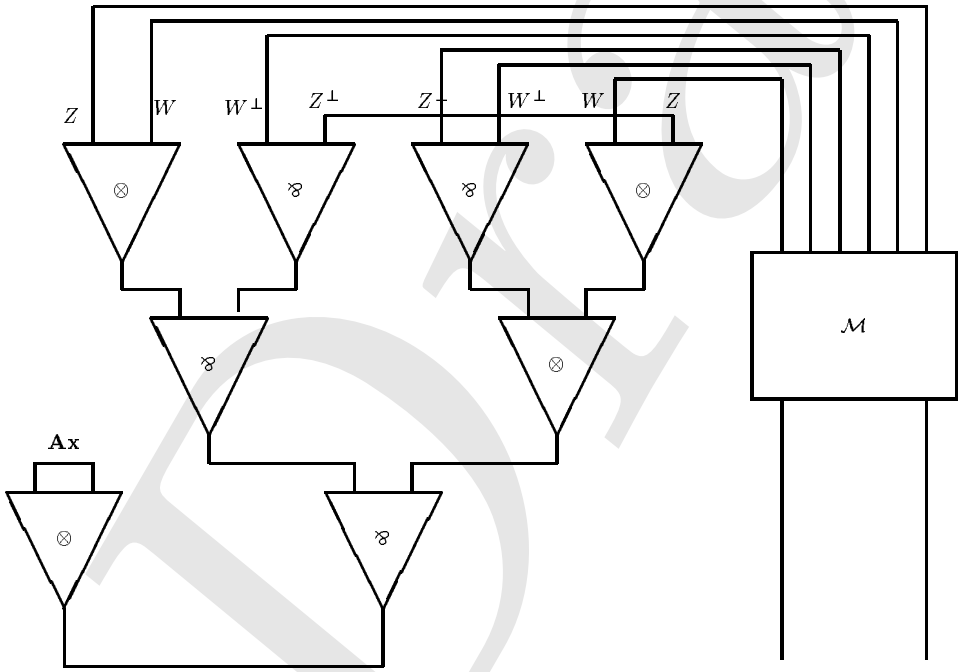
Here is a small example for this construction. The proof-structure below is not correct because for one of the switching, the graph has a cycle:



Then we form a module which interact with the previous proof-structure by creating a dead-lock. This module is correct and can be completed to have a proof-net:



The connection of the bad proof-structure and the proof-net reduce to a dead-lock:



The conjonction of the two lemmas gives a proof of the main theorem on the minimality of the correctness criterion.

5. Correct Modules

This technical section gives proofs of several properties on modules (Danos and Regnier, 1989). Modules are extension of proof-structures where some formulas can act as premises (proof-structure has only conclusions). The *frontier* of a module is the multi-set of its conclusion formulas and its premise formulas. There exists a criterion for modules that distinguishes correct modules. The correctness criterion says that such a module is a part of a proof-net.

Definition 16. A *correct module* is a module which is a sub-module of a proof-net.

Theorem 17. A module is correct if and only if for any switching, the corresponding graph:

— has no cycle,
— has no internal component (every component must be connected to one of the formulas of the frontier of the module).

This is the *correctness criterion* for modules.

Proof.
If we have a correct module then it is a sub-module of a proof-net. The graph corresponding to a switching has no cycle and is connected for this proof net. Thus, the graph induces by a switching on the module can not have any cycle and can not have an internal component because such a configuration will create a cycle or an internal component on the proof-net which is impossible.
The converse is not immediate. The proof is obtained by recurrence on the number of conclusions and premises of the modules. The proof needs some intermediate results.
Firstly, $\mathcal{M}$ is a module that verifies the criterion for modules if and only if the module obtained from $\mathcal{N}$ by adding an axiom link for each premise of $\mathcal{M}$ verifies the criterion for modules. This property is also true for correct modules. Thus, we can only work with proof-structure (module with no premise).
Secondly we must establish the following lemma: if $\mathcal{M}$ is a proof-structure that verifies the criterion for module then either every switching gives a connected graph or there exists two conclusions A and B such that for any switching they are not connected. This property is very similar to the property of proof-net to be saturated. In fact, suppose that for any conclusions A and B, there exists a switching such that the two formulas are in the same component, then the proof-structure is saturated. This means that for any $\parr$-link of the proof-structure, there exists a switching such that the two premises of this $\parr$-link are connected. This fact results from the fact that for any switching there are no internal component and that there exists a switching that connects each couple of conclusions. If the proof-structure is not saturated, there exists a $\parr$ such that its premises are not connected. But both are connected to two different conclusions (no internal component). There exists a switching such that those two premises are connected. If we look at the three paths, since there are no path between the premises of the $\parr$-link, we can find another internal path that must be connected to a third conclusion (no internal path). By alternating the conditions that say that two conclusions must be connected by a path

(for a switching) and that there are no internal path, we build an endless list of different conclusions. This is impossible since there are a finite number of conclusion.
Thus we conclude that if $\mathcal{M}$ is a proof-structure that verifies the criterion for modules then either $\mathcal{M}$ is a proof-net or there exists two conclusions A and B such that for any switching they are not connected.
Now the proof follows because if we have a proof-structure that verifies the criterion for module then either $\mathcal{M}$ is a proof-net and so it is a correct module either we can construct a proof-structure with less conclusions by connecting the two conclusions such that for any switching they are not connected by a new $\otimes$-link. This new proof-structure has no cycle and has no internal component for any switching. □

Theorem 18. Let $\mathcal{M}$ be a module without cut and with only one conclusion. If $\mathcal{M}$ is correct then we can find another module without cut and with the dual premise of $\mathcal{M}$ such that the proof-structure obtained by connecting the premises by axioms is a proof-net.

Proof. We need to change the proof of the previous theorem by adding the fact that there is always a path between the conclusion of $\mathcal{M}$ and the other formulas of its frontier. □

6. Conclusion

The result presented here may be used for linear logic with the Mix rule (Fleury and Retoré, 1994) or for interaction nets (Lafont, 1990). The theorem remains the same except that the basically wrong configurations are only the dead-lock (and not a disconnected proof-structure). The proof is simpler because here the correctness criterion for module only says that there must be no cycle for any switching and due to the fact that a sub-proof-structure of a proof-net is automatically a proof-net.

One may wander if we can extend this result to other fragments of linear logic. The answer is not obvious. In particular, it is difficult to define the commutative-conversions for the additives since we need to define the notion of empire that is not defined on any proof-structure (only on proof-net). This is also the case for the constant $\bot$ and for the weakening for plain linear logic. However, with the Mix rule, a reducing mechanism without box may be defined for multiplicatives and exponentials with constants. However, there exists some counter-examples that show us that the correctness criterion for this fragment is not minimal. Another negative result comes from the second order quantification. Because a cut may be emulated by an *it exists*, it is not a surprise to show that the correctness criterion for this fragment is not minimal.

However, we have demonstrated here that for the multiplicative fragment, the correctness criterion is minimal for proof-structure without cut if we want that cut-elimination does not lead to the very bad configurations: dead-lock or disconnected proof-structure. We are looking for different extensions for the other fragments (exponentials in particular) with the same property.

References

Danos, V. (1990). La Logique Linéaire Appliquée à l'Étude de Divers Processus de Normalisation at Principalement du Lambda-Calcul. Thèse de Doctorat, Université de Paris VII.

Danos, V. and Regnier, L. (1989). The structure of multiplicatives. *Archive for Mathematical Logic*, 28:181–203.

Fleury, A. and Retoré, C. (1994). The mix rule. *Mathematical Structures in Computer Science*, 4(2).

Girard, J.-Y. (1987). Linear logic. *Theoretical Computer Science*, 50:1–102.

Lafont, Y. (1990). Interaction nets. In *Proc. 17-th ACM Symp. on Principles of Programming Languages, San Francisco*, pages 95–108.