A by-level analysis of Multiplicative Exponential Linear Logic

Marco Gaboardi[1], Luca Roversi[1], and Luca Vercelli[2]

[1] Dipartimento di Informatica - Università di Torino
[2] Dipartimento di Matematica - Università di Torino
http://www.di.unito.it/~{gaboardi | rover | vercelli}

Abstract. We study the relations between Multiplicative Exponential Linear Logic (meLL) and Baillot-Mazza Linear Logic by Levels (mL^3). We design a decoration-based translation between propositional meLL and propositional mL^3. The translation preserves the cut elimination. Moreover, we show that there is a proof net Π of second order meLL that cannot have a representative Π' in second order mL^3 under any decoration. This suggests that levels can be an analytical tool in understanding the complexity of second order quantifier.

1 Introduction

The implicit characterization of the polynomial and elementary time computations by means of structural proof theory takes its origins from a predicative analysis of non termination. We recall, indeed, that Girard conceived Elementary Linear Logic (ELL) and Light Linear Logic (LLL) [1] by carefully analyzing the formalization of naïve set theory inside the Multiplicative and Exponential fragment of Linear Logic (meLL). The comprehension scheme could be represented without any paradoxical side effect by forbidding the logical principles *dereliction* $!A \multimap A$ and *digging* $!!A \multimap !A$.

Intuitively, without dereliction and digging the proof nets of both ELL and LLL are *stratified*. Namely, during the cut elimination process, every node of a proof net either disappears or it is always contained in a constant number of *regions*, called boxes. The stratified proof nets of ELL are characterized by a cut elimination cost which is bounded by an elementary function whose parameters are the size of a given net Π and its depth, *i.e.* the maximal number of nested boxes in Π.

Moreover, Girard noted that ruling out the *monoidality* of the functor "!", i.e. $(!A \otimes !B) \multimap !(A \otimes B)$, from ELL yields LLL whose cut elimination cost lowers to a polynomial. The reason is that the logical connective $\otimes$ somewhat allows to count the resources we may need. Commuting $\otimes$ with ! hides the amount of used logical resources because of the *contraction* $!A \multimap (!A \otimes !A)$. So, the absence of monoidality allows to keep counting the needed resource by means of $\otimes$.

In [2], the authors pursue the predicative analysis on meLL by introducing mL^3. This system generalizes ELL by means of explicit indices associated to the edges of the proof nets of meLL. Moreover, further structural restrictions on mL^3 yield a polynomial time sound generalization mL^4 of LLL. The use of indices in meLL analysis is not new and traces back to, at least, [3, 4]. The new systems mL^3 and mL^4 still characterize implicitly the elementary and polynomial computations. Their distinguishing feature lies in a more flexible use of the nodes that change the level, so, somewhat generalizing the notion of box.

Since mL^3 is a restriction of meLL, it is natural that some derivation of the latter cannot be represented inside the former, the reason being we know that the cost of the cut elimination of meLL overwhelms the elementary one of mL^3.

In this paper, we show that indices strongly restrict meLL proof nets in presence of $\exists$ and $\forall$, while they are minor restriction when quantifier do not get used. Indeed, we can show that every proof net Π in the *propositional fragment* of meLL has a representative Π' in mL^3 that preserves the cut elimination. Specifically, Π' is the result of a predicative analysis of Π, based on indices we can use to label every edge of mL^3 proof nets. The proof net Π' of mL^3 is the result of the algorithm @, applied to Π, we introduce in this work. The proof net @(Π) is a decoration of both the edges and the formulæ of Π, using the paragraph modality §, whose instances correspond to an index change in the proof net of mL^3 being constructed.

The interest of the translation that @ implements is twofold. Concerning the structural proof theory, @ shows that the modality § internalizes the notion of index at the level of the formulæ. Concerning the implicit characterization of complexity classes, @ offers the possibility of a finer study of normalizations measures of propositional meLL, thanks to the structural aspects that mL^3 supplies.

Finally we answer negatively to the following two natural questions: (i) Is there any extension of @ able to translate every proof net of (full) meLL into mL^3?, and (ii) Is there any translation, alternative to any generalization of @, from meLL to mL^3? The reason of the negative answer lies in the proof net Π of meLL in Figure 6. There is no decoration Π' in mL^3 of Π because to obtain Π' either we should collapse two distinct indices of Π' or we would need a new node able to change indices but not the formulæ. Both solutions would imply a cut elimination cost blow up, unacceptable inside mL^3.

Summing up, the predicative analysis of meLL by means of the indices inside mL^3 identifies as the true source of impredicativity of meLL *the collapse of indices, implicit in the second order quantification of the formulæ of* meLL *itself*. Then, the "side effect" of such a collapse is the huge cut elimination bound of meLL.

Acknowledgments. We warmly thank the anonymous referees for the detailed comments and suggestions on the preliminary version of the paper.

2 Second order meLL

We start by recalling second order Multiplicative Exponential Logic (meLL) in proof nets style. In particular, analogously to [2], we present a meLL version including the paragraph (§) modality.

The formulæ. meLL derives multisets of formulæ that belong to the language generated by the following grammar:

$$F ::= A \mid \flat A \qquad A ::= \alpha \mid A \otimes A \mid A \parr A \mid \forall \alpha.A \mid \exists \alpha.A \mid !A \mid ?A \mid \S A \mid A^{\perp}$$

The start symbol F generates both *(standard) formulæ* and *partially discharged formulæ*. Standard formulæ are generated from the start symbol A. Partially discharged formulæ are of kind $\flat A$; the syntax prevents nesting of $\flat$ symbols. We shall use A, B, C, possibly with sub or superscripts, to range over standard formulæ, F, G to range over formulæ. Γ, Δ, Ξ range over multisets of formulæ. The standard meLL formulæ are quotiented by the De Morgan rules, where $(A, A^{\perp})$, $(\otimes, \parr)$, $(\forall, \exists)$, $(\S, \S)$ and $(!, ?)$ are the pairs of dual operators. Notice that § is self dual, namely: $(\S A)^{\perp} = \S(A^{\perp})$.

Proof nets of meLL. Given the nodes in Figure 1, we say that an *Axiom node is a proof net*. Moreover, given two proof nets:

$\Pi_1 : F_1, \ldots, F_m \qquad \Pi_2 : G_1, \ldots, G_n$

with $m, n \geq 1$, then all the graphs inductively built from Π_1 and Π_2 by the rule schemas in Figure 2 are proof nets.

Cut elimination in meLL. Every pair of dual *linear* nodes (axiom/cut, $\otimes/\parr$, $\forall/\exists$, §/§) annihilates in one step of reduction, as usual in literature. The *exponential* pair of dual nodes !/? rewrites by means of the *big-step* in Figure 3.

Basic definitions and properties in meLL. The modality § is not part of the original version of meLL; it is easy to show that in meLL § is, essentially, useless, i.e. A and $\S A$ may be proved equivalent in meLL. Nevertheless, § become useful when handling sublogics of meLL. $\S^k A$ means $\S \ldots \S A$ with k paragraphs.

The original formulation of meLL also contains the **mix** rule and **units**, but for simplicity we omit them.

A **weakening** node is a contraction with 0 premises. We call **axiom-edge**, **weakening-edge**, **cut-edge**, etc. an edge connected to an an axiom node, a weakening node, a cut node, etc..

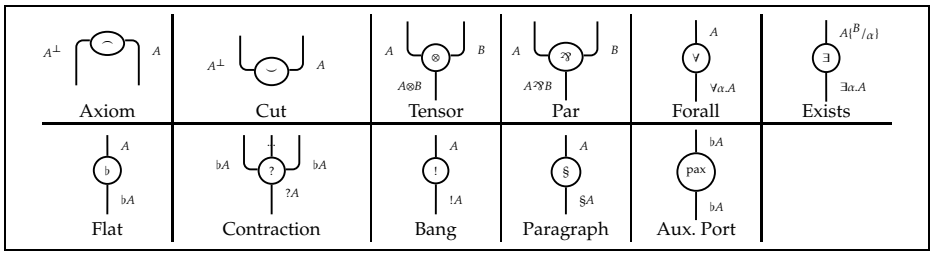

Fig. 1. Nodes for the nets of meLL.

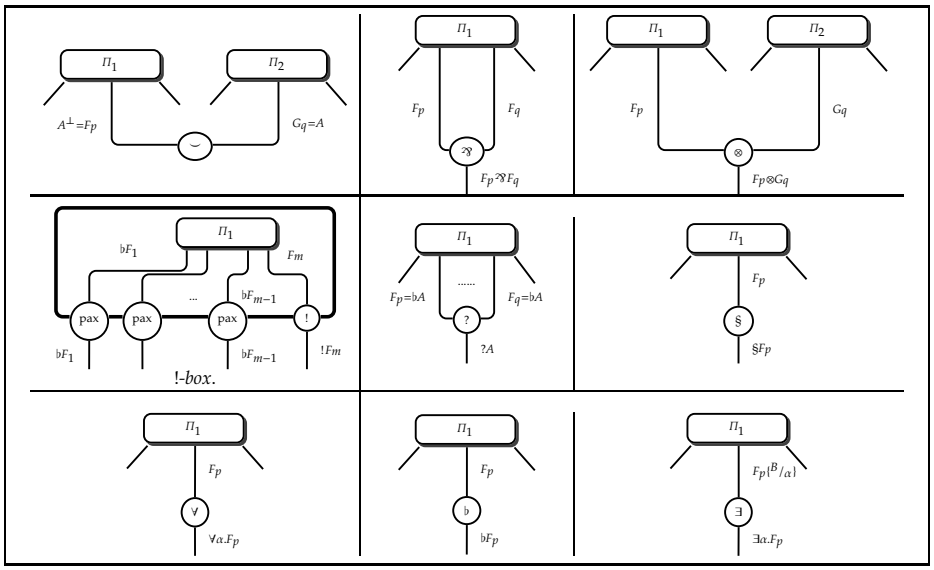

Fig. 2. Inductive rule schemes to build proof nets of meLL.

Fact 1 (About the Structure of the Proof Nets) *Let Π be a proof net of* meLL*, and u one of its cut links or conclusions. Let ρ be a graph-theoretical path along Π from u to an axiom or to a weakening node v, not containing any other axioms. Then ρ does not contain any other cut node.*

Thanks to this Fact, we can state that all the edges of our proof nets are directed downwards, from axioms or weakening nodes towards conclusions or cut nodes, even if we do not draw the corresponding arrows. A **path** inside a meLL proof net Π is a sequence of nodes $\tau = \langle u_0, \ldots, u_k \rangle$ in Π such that (i) each u_i is connected with u_{i+1}, (ii) the direction of such edge is from u_i towards u_{i+1}, and (iii) for every i, $u_i \neq u_{i+1}$. The **size** of a meLL proof net is the number of its nodes.

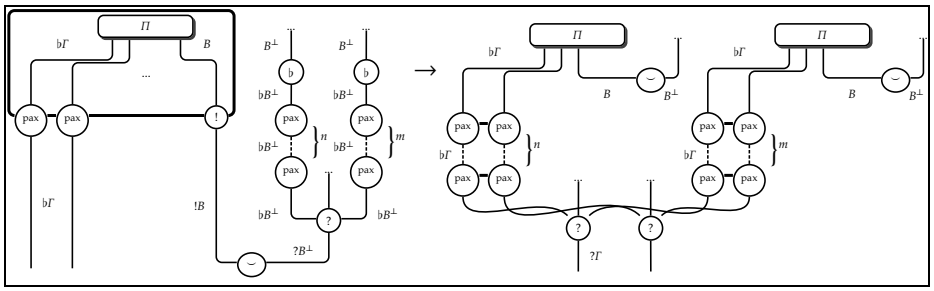

Fig. 3. Big-step reduction. A contraction with k premises in the redex implies k copies of Π in the reduct. For the sake of clarity we do not draw all the boxes in the picture.

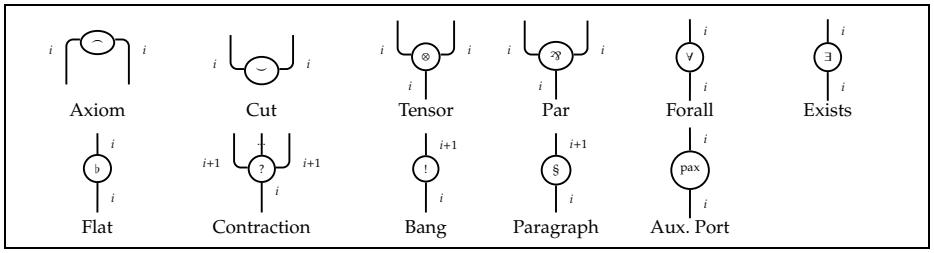

Fig. 4. Costraints for indexing meLL proof nets.

3 Multiplicative Linear Logic by Levels: mL^3

The system mL^3 is described in [2]. It is the subsystem of all the proof nets of meLL admitting an *indexing*:

Definition 1. *Let Π be a proof net of* meLL*. An* **indexing** *for Π is a function I from the edges of Π to $\mathbb{Z}$ satisfying the constraints in Figure 4 and such that $I(e) = I(e')$ for all the conclusions e, e' of Π.*

Fact 2 (Indexes do not Increase from Axioms to Conclusions) *Let Π be an* mL^3 *proof net, I an indexing for Π, ρ a path from some node u to some node v. Then $I(u) \geq I(v)$.*

It will be convenient to consider a particular kind of indexing.

Definition 2. *Let Π be an* mL^3 *proof net, and I be an indexing for Π. We say that I is* **canonical** *if Π has an edge e such that $I(e) = 0$, and $I(e') \geq 0$ for all edges e' of Π.*

Fact 3 (Existence of Canonical Indexing [2]) *Every proof net of* mL^3 *admits one and only one canonical indexing.*

We can now define a measure on mL^3 proof nets.

Definition 3. *Let Π be an mL^3 proof net, and let I_0 be its canonical indexing. The* **level** *of Π is the maximum integer assigned by I_0 to the edges of Π.*

If 2^n_x is the function such that $2^n_0 = 2^n$ and $2^n_m = 2^{2^n_{m-1}}$, then:

Theorem 1 (Elementary bound for mL^3 [2]). *Let Π be an mL^3 proof net of size s and level l. Then, the round-by-round cut-elimination procedure reaches a normal form in at most $(l+1)2^s_{2l}$ steps.*

The Theorem above is a result of *weak* polynomial soundness, as it only has been proved for a particular cut-elimination procedure. It is reasonable however that it can be generalized to any reduction strategy, in analogy to what happens in ELL and LLL [5]. The interested reader may find the definition of the round-by-round procedure and a proof of Theorem 1 in [2].

4 Embedding propositional meLL into mL^3

Definition 4. *Let Π be a proof net of meLL. A* **quasi-indexing** *for Π is a function Q from the edges of Π to $\mathbb{Z}$ that respects all the constraints in Figure 4, with the possible exception of the axiom edges, and such that for all conclusion e, e' of Π it holds $Q(e) = Q(e')$.*

Fact 4 (Quasi-Indexing Exists) *Every meLL proof net admits a quasi-indexing.*

Proof. Let Π be a proof net of meLL; we want to build some quasi-indexing Q. We call $c_1, \ldots, c_n$ the cut nodes of Π. We arbitrarily choose a value $Q(e) = i$ for all the conclusion edges e of Π, and a value $Q(e^j_1) = Q(e^j_2) = i_j$ for every couple of edges e^j_1, e^j_2 incident in c_j. Then, using the rules in Figure 4, we can calculate the value of Q in all the edges of the proof net. The process of calculation terminates when the axiom and weakening nodes are reached. □

For every Π, whose cut nodes are $c_1, \ldots, c_n$, we call $Q(i, i_1, , ...i_n)$ the (unique) quasi-indexing that has value i on the conclusions and value $i_1, \ldots, i_n$ on the cut-edges. This definition is justified looking at the proof of Fact 4.

The coming *level of a formula* is completely unrelated to the levels of Definition 3:

Definition 5. *For every formula A of meLL let the* **formula level** $\mathtt{fl}(A)$ *be:*

$$\mathtt{fl}(\alpha) = 0 \qquad \mathtt{fl}(\diamond A) = \mathtt{fl}(A) + 1 \qquad \diamond \in \{!, ?, \S\}$$
$$\mathtt{fl}(\flat A) = \mathtt{fl}(A) \qquad \mathtt{fl}(A \,\square\, B) = \max\{\mathtt{fl}(A), \mathtt{fl}(B)\} \qquad \square \in \{\otimes, \parr\}$$

Definition 6. *Let Π be a proof net of meLL, Q a quasi-indexing for it. Let e be an edge in Π, labelled by a formula A. Then, the* **absolute level** *of e in Π is defined as $\mathtt{al}(e) = Q(e) + \mathtt{fl}(A)$.*

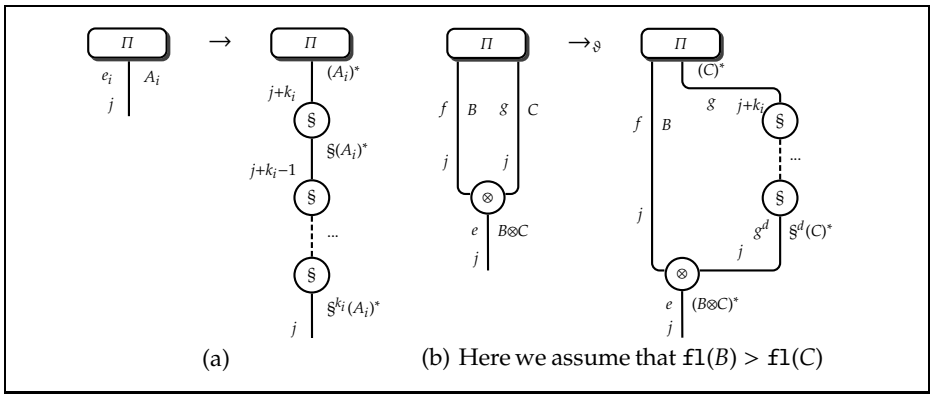

Fig. 5. The main cases of the rewriting steps performed by @ (Proof of Proposition 1).

Notice that the definition depends on the chosen quasi-indexing.
The following map is crucial in the proof of Proposition 1:

Definition 7. *For every* meLL *formula A let* $(A)^*$ *be defined as:*

$$
\begin{aligned}
(\alpha)^* &= \alpha \\
(B \,\square\, C)^* &= §^d(B)^* \,\square\, (C)^* \quad \text{if } d = \mathtt{fl}(C) - \mathtt{fl}(B) \geq 0 \qquad \square \in \{\otimes, \parr\} \\
(B \,\square\, C)^* &= (B)^* \,\square\, §^{-d}(C)^* \quad \text{if } d = \mathtt{fl}(C) - \mathtt{fl}(B) \leq 0 \qquad \square \in \{\otimes, \parr\} \\
(\diamond A)^* &= \diamond((A)^*) \qquad \diamond \in \{!, ?, \flat\}\,.
\end{aligned}
$$

The algorithm @. The main result of this section concerns the following algorithm @. Let the arguments of @ be a proof net $\Pi : A_1, \ldots, A_n$ of propositional meLL and a quasi-indexing Q for Π. The algorithm returns an mL^3 proof net. We will give a direct proof of this fact. Let the conclusions and the cut edges of Π be $e_1, \ldots, e_n$. Let $K = \max_{1 \leq i \leq n} \{\mathtt{al}(e_i)\}$. For every edge e_i, with $1 \leq i \leq n$, labelled with the formula A_i, we define @ to perform the following steps:

1. Replace A_i by $(A_i)^*$.
2. Add k_i new (§) nodes after the edge e_i where $k_i = K - \mathtt{al}(e_i)$, label the new edges respectively by $§^1(A_i)^*, \ldots, §^{k_i}(A_i)^*$ and modify the quasi-indexing accordingly. Note that now $\mathtt{al}(e_i) = K$. See Figure 5(a).
3. Apply the subroutine ϑ of @, here below, to the edge e_i.

The subroutine ϑ takes an edge e of (the already modified version of) Π as its argument. ϑ is recursive and is defined by cases on the kind of the edge e:

(a) If e is an axiom edge, then it is done.
(b) If e is the conclusion of a (⊗) node with premises the edges f and g labelled with formulae B and C respectively, then replace B by $(B)^*$ and C by $(C)^*$

respectively. Let us suppose for clarity that $\mathtt{fl}(B) > \mathtt{fl}(C)$ (see Figure 5(b)). Calling $d = (\mathtt{fl}(B) - \mathtt{fl}(C))$, we add d new (§) nodes after the edge g and label the new edges $g^1, \ldots, g^d$ respectively by $\S^1(C_i)^*, \ldots, \S^d(C_i)^*$. Modify Q accordingly, then apply ϑ on f and g.

(c) If e is the conclusion of a (!) node (or (♭) or (pax)) with premises the edge f labelled with the formula B then replace it by $(B)^*$ and apply ϑ on f.

(d) If e is the conclusion of a (?) node with premises the edges $f_1, \ldots, f_l$ labelled with formulae $B_1, \ldots B_l$, then replace them by $(B_1)^*, \ldots, (B_l)^*$ and apply ϑ on every $f_1, \ldots, f_l$.

Proposition 1 (Embedding Propositional meLL into mL^3). *There is an algorithm* $@(\cdot,\cdot)$ *that takes every proof net* Π *of propositional* meLL*, endowed with a quasi-indexing* Q*, and returns a proof net* $@(Q,\Pi)$ *of* mL^3*. The proof nets* Π *and* $@(Q,\Pi)$ *only differ for the possible presence of some new paragraph nodes.*

Proof. $@(\cdot,\cdot)$ is the algorithm already described. $@(\cdot,\cdot)$ transforms a proof net Π of meLL in a new graph $@(Q,\Pi)$, with conclusions labelled by $\S^{k_1}(A_1)^*, \ldots, \S^{k_n}(A_n)^*$, for some $k_1, \ldots, k_n$, to which it is naturally associated a quasi-indexing Q'. The quasi-indexing Q' associates to conclusions and cut edges of $@(Q,\Pi)$ the same indices as Q assigns to conclusions and cut edges of Π. We need to check that $@(Q,\Pi)$ is really a proof net of meLL, and that this proof net is in mL^3.

Let us consider the transformations previously described. The untyped graph is still an untyped proof net of meLL, because we have just added some paragraphs. Moreover, by construction every edge e of Π labelled by A is translated into an edge e' of $@(Q,\Pi)$, labelled by $(A)^*$. So, in particular, axioms, cuts and contractions are labelled correctly. The labelling of the other nodes follows by construction of $@(\cdot,\cdot)$.

At last, we need to show that Q' is an indexing. Let us consider two edges f, g incident into an axiom in $@(Q,\Pi)$, labelled resp. by A and $A^\perp$. Notice that, by construction, for every edge e of $@(Q,\Pi)$ it holds $\mathtt{al}(e) = K$. As a consequence, f and g have the same quasi-index $Q'(e) = \mathtt{al}(e) - \mathtt{fl}(A) = K - \mathtt{fl}(A)$, and so Q' is also an indexing. □

Proposition 2 ($@(\cdot,\cdot)$ preserves the Cut-Elimination). *For every reduction* $\Pi \to^+ \Sigma$ *in propositional* meLL*, and for every quasi-indexing* Q *of* Π*, there exists a quasi-indexing* $\tilde{Q}$ *of* Σ *such that* $@(Q,\Pi) \to^+ @(\tilde{Q},\Sigma)$*:*

$$\begin{array}{ccc} \Pi & \to^+ & \Sigma \quad \textit{in meLL} \\ \downarrow & & \downarrow \\ @(Q,\Pi) & \to^+ & @(\tilde{Q},\Sigma) \quad \textit{in mL}^3 \end{array}$$

Proof. It is enough to prove the result for 1-step reductions $\Pi \to \Sigma$. So, let c be the cut fired during this reduction; c corresponds to a unique cut c' of $@(Q,\Pi)$. By construction of $@(\cdot,\cdot)$, the only difference between Π and $@(Q,\Pi)$ is the possible presence of paragraphs. As many (§) nodes as T may occur just above c'. If we eliminate all the T (§) nodes we have that the edges entering c' correspond to

the edges entering c. Firing c' yields a proof net Θ of mL^3. We have to show that $\Theta = @(\tilde{Q}, \Sigma)$, for some $\tilde{Q}$. If c was a cut with an axiom, or a cut between a weakening and a closed box, then both c and c' annihilate. Otherwise, we get (at least) one residual c'' of c' inside Θ. We can define $\tilde{Q}$ equal to Q on all the conclusions and cut edges that are not involved in the reduction, and that is defined on the edges entering c'' as follows. We distinguish two cases. If c is not an exponential cut, e is an edge incident to c, and f is an edge incident to c'', then $\tilde{Q}(f) = Q(e) + T$. If c is an exponential cut, $\tilde{Q}(f) = Q(e) + T + 1$. □

Corollary 1 (Complexity Bound for meLL). *Let Π be a proof net of* meLL*. Let's call $M = \max\{\mathtt{fl}(A) \mid A$ a formula labelling an edge of $\Pi\}$. Then, the round-by-round cut-elimination procedure of Π terminates in at most $(M+1) \cdot 2_{2M}^{M \cdot |\Pi|}$ steps.*

Proof. Let us fix the quasi indexing $Q = Q(0, 0, \ldots, 0)$, and let us calculate $@(Q, \Pi)$. Notice in particular that (i) the constant $K = \max_{1 \leq i \leq n} \{\mathtt{al}(e_i)\}$ used defining @ in this case is $K = \max_{1 \leq i \leq n} \{\mathtt{fl}(e_i)\} \leq M$; and (ii) the indexing I induced on $@(Q, \Pi)$ is canonical. We want apply Theorem 1 to $@(Q, \Pi)$. The size $|@(Q, \Pi)|$ is bounded by $K \cdot |\Pi|$: indeed, for every node of Π, @ adds at most K new (§) nodes. The level of $@(Q, \Pi)$ is $l = \max\{I(e) \mid e$ is an edge of $@(Q, \Pi)\}$. Every $I(e)$ is bounded by K, so $l \leq K$. Thus, $@(Q, \Pi)$ reduces in at most $(K+1) \cdot 2_{2K}^{K \cdot |\Pi|} \leq (M+1) \cdot 2_{2M}^{M \cdot |\Pi|}$ steps because of Theorem 1. At last, Proposition 2 tells that Π reduces in at most as many steps as $@(Q, \Pi)$, and the thesis follows. □

5 The full meLL case

The Proposition 1 fails for second order meLL proof nets. The counterexample is the proof net Π in Figure 6. The behaviour of Π is analogous to the λ-term $(\lambda x.xx)\overline{2}$. Note that the argument $\overline{2}$ of Π is not really necessary, but it is makes evident the dynamic interaction of the two occurrences of x.

We call ρ the path starting from the axiom **v** and arriving into the contraction **u** passing through the (∃) node $\mathbf{w_1}$; we call τ the path starting from **v** and arriving into **u** passing through the (∃) node $\mathbf{w_2}$.

Firstly, we can imagine to extend the algorithm @ used in the proof of Proposition 1, to a new algorithm $\overline{@}$. It is necessary to extend the definitions of the map $(\cdot)^*$ and of the *formula level*. The most naïve assumption is that $(QA)^* = Q(A)^*$ and $\mathtt{fl}(QA) = \mathtt{fl}(A)$ for each quantifier Q. It will be enough to study the behaviour of $\overline{@}$ along the paths ρ and τ. Starting from the cut node **c**, $\overline{@}$ would add several new (§) nodes to Π, in particular over the *right* premise of the (⊗) nodes $\mathbf{z_1}$ and $\mathbf{z_2}$, but no new nodes over ρ and τ. So, the resulting net would not admit any indexing, because the two edges incident in **v** would still have different quasi-indices 2 and 3.

Now, the reader may legitimately think that this problem is due to our particular (and naïve) definition of the algorithm $\overline{@}$. In fact, the problem is more serious.

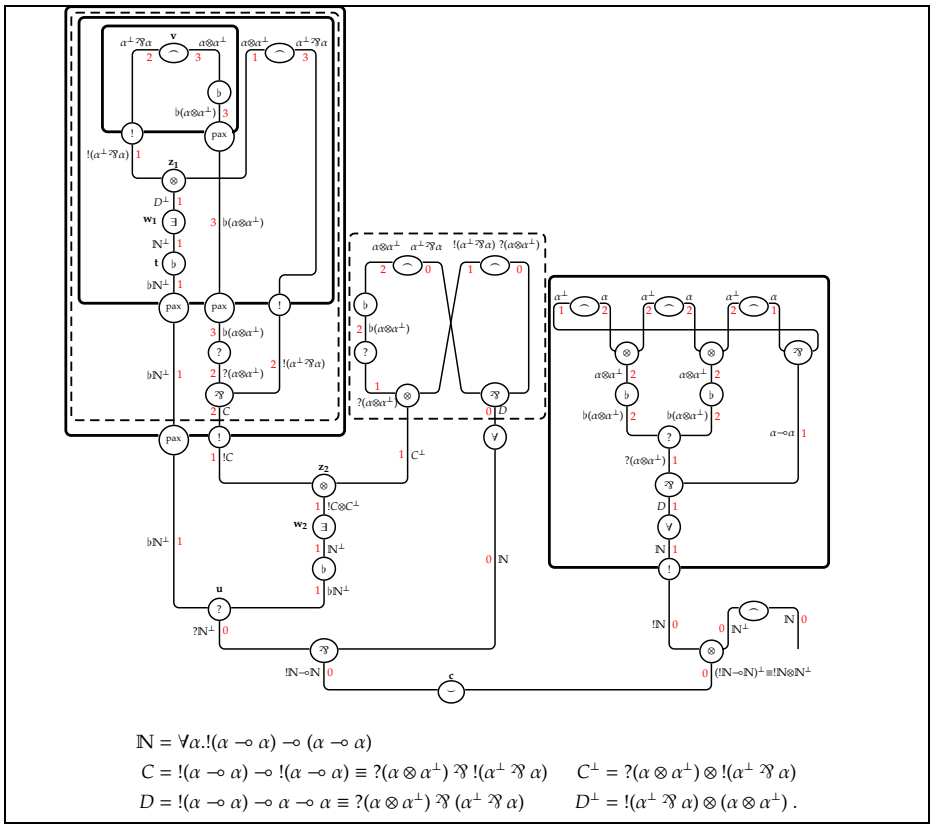

Fig. 6. This proof net represents the λ-term $(\lambda x.xx)\overline{2}$. The two dashed boxes are the proof nets proving $\vdash C^\perp, D$ and, essentially, $\vdash ?D^\perp, C$.

We will show that *there is no way of building an* mL^3 *proof net just adding some* (§)*-nodes to* Π. In order to have the same index on the two sides of **v**, we need to add along the path ρ one (§) node more than the ones we add along τ. The problem arises as the two formulas labelling the premises of **u** must be equal. Along ρ, a (§) node can be added only along the four edges connecting **v** to **t**; but whatever edge we choose, if we add a (§) node along it, we are forced to add another (§) node along τ to make the premises of **u** agree. And so the resulting proof net cannot be indexed.

6 Concluding Remarks and Further works

The main contribution of our work is a predicative analysis of meLL by means of the indices inside mL^3. Such an analysis highlights that the source of the huge complexity cost of meLL is due to the use of second order quantifiers that

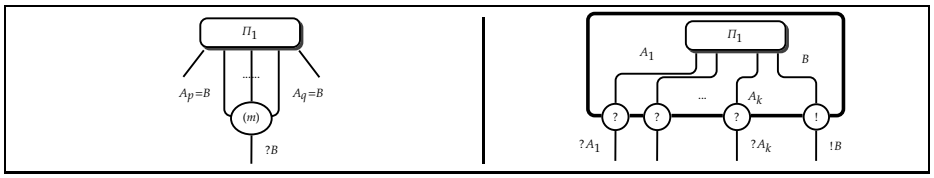

Fig. 7. Exponential inductive rule schemes to build proof nets of SLL.

hide and collapse indices. Our analysis is also connected to other problems, that motivate some further developments we outline in the following.

mL^3 as a framework for ICC. We recall that the main reason behind mL^3 is to better understand computations with elementary cost. This work is to support the idea that mL^3 is very useful to characterize other complexity classes. Of course, the simple definition of mL^4 as a subsystem of mL^3, that generalizes a simplified version of LLL, studied in [6,7], already supports such an idea. We strengthen it further by embedding the propositional fragment of SLL [8] in mL^3. We recall that the formulæ of SLL are a subset of the meLL ones. The proof nets of SLL are built using the "linear" nodes of meLL, and the "exponential" nodes in Figure 7. Our embedding of SLL into mL^3 is based on an intermediate embedding of SLL into meLL. Let us call **exponential** every path from a (♭) node u of a meLL proof net to the first (?) node we may cross, starting from u. SLL can be identified with the subsystem of meLL including all and only the proof nets Π that satisfy the following conditions:

R1: Every exponential path entering a (?) node with one premise crosses at most one (pax) node.
R2: Every exponential path entering a (?) node with more that one premise does not cross any (pax) node.
§N: No (§) node occurs in Π.

R1 and **R2** simplify analogous conditions in [7]. Basing it on the **R1**, **R2**, and **§N**, we define the following map algorithm from the proof nets of SLL to those ones of meLL. Every (?) node of Π becomes a (♭) node followed by a (pax) node followed by a (?) node. Every multiplexor (m) with k premises becomes a tree composed by k (♭) nodes, followed by a (?) node. Proposition 1 implies that propositional SLL has a corresponding subsystem in mL^3. In particular, it is easy to verify that such a subsystem is the one obtained by considering only the proof nets of propositional mL^3 satisfying exactly **R1** and **R2** since @ preserves them.

Our future work is on the embedding of full SLL into mL^3. This should be possible because the structural constraints that lead from meLL to SLL limit the interaction between second order quantifiers and indices, implicitly hidden by the of-course modality. The proof net in Figure 6, not in SLL, supports this

idea, because the second order quantifiers, associated to the duplication-related modality, may require to collapse indices which must be necessarily distinct, as already observed in Section 5.

Complexity bounds for the simply typed λ-calculus. We also aim at a proof theoretical based analysis of the computational complexity of the simply typed λ-calculus, which, under the Curry-Howard analogy, can correspond to intuitionistic propositional meLL. We mean we want to trace back to simply typed λ-calculus the purely structural analysis of the computational complexity that mL^3 supplies for propositional meLL. The point is to avoid any reference to the type of a given simply typed λ-term to infer its normalization cost, as in [9, 10]. First steps in this direction are Proposition 1, and a careful inspection of the definition of @. Let Π be a proof net of propositional meLL. Proposition 1 implies that the length of the reduction sequences of @(Π) in mL^3 bound those ones of Π. The definition of @ reveals a relation between the structure of Π and the level of @(Π). The latter comes from the formulæ levels of formulæ of *only specific axiom nodes* of Π. So, the open points for coming work are at least two: (i) Is there any linear or polynomial function relating the size of Π and the level of @(Π)?, and (ii) Is there any alternative @′ to @ never using the formulæ of the above specific axioms in Π able to yield @′(Π) in mL^3?

References

1. Girard, J.Y.: Light linear logic. Inf. Comput. **143**(2) (1998) 175–204
2. Baillot, P., Mazza, D.: Linear logic by levels and bounded time complexity. Accepted for publication in Theor. Comp. Sci. (2009)
3. Martini, S., Masini, A.: On the fine structure of the exponential rule. In Girard, J.Y., Lafont, Y., Regnier, L., eds.: Advances in Linear Logic. Cambridge University Press (1995) 197–210 Proceedings of the Workshop on Linear Logic.
4. Martini, S., Masini, A.: A computational interpretation of modal proofs. In Wansing, H., ed.: Proof Theory of Modal Logic. Volume 2., Dordrecht (1996) 213–241
5. Terui, K.: Light affine lambda calculus and polynomial time strong normalization. Arch. Math. Logic **46**(3-4) (2007) 253–280
6. Mairson, H.G., Terui, K.: On the computational complexity of cut-elimination in linear logic. In: ICTCS. (2003) 23–36
7. Mazza, D.: Linear logic and polynomial time. Mathematical Structures in Computer Science **16**(6) (2006) 947–988
8. Lafont, Y.: Soft linear logic and polynomial time. Theor. Comp. Sci. **318** (2004) 163–180
9. Schwichtenberg, H.: Complexity of normalization in the pure typed lambda-calculus. In Troelstra, A.S., van Dalen, D., eds.: Proc. of Brouwer Centenary Symp. Volume 110 of Studies in Logic and the Foundations of Math. North-Holland (1982) 453–457
10. Beckmann, A.: Exact bounds for lengths of reductions in typed lambda-calculus. J. Symb. Log. **66**(3) (2001) 1277–1285