Do Light Logics allow a unified view of Stratification and Boundedness?

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

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

Abstract. This few pages are meant to illustrate an ongoing work whose goal is to find fine-grained logical principles, in the tradition of structural proof-theory, on which we can base the definition of a deductive system that contains both Light Affine Logic and Soft Linear Logic.

1 Introduction

1.1 Bounding the computational cost inside the structural proof theory

One of the key features of Linear Logic (LL) is that the structural rules, which account for duplication, hence for the complexity of the cut elimination, are only allowed on "modal" formulæ. The study about the cost of the cut elimination in Linear Logic may be done using proof nets, a graphical representation of LL proofs. In the particular version of LL we consider, proof nets introduce modal formulæ by means of special constructs, called "functorial boxes", "digging" and "dereliction", and only the proof nets inside a box may be duplicated during the cut elimination process.

Some constraints can be imposed on LL in order to achive bounds on the number of steps nedded to perform the cut elimination. Forbidding the use of digging and dereliction leads to Elementary Linear Logic (ELL) [Gir98]. ELL enjoys an elementary bound on the number of duplications that can occur during the cut elimination of any of its derivations. Through the Curry-Howard correspondence, ELL characterizes the class of elementary functions. Recall that they are those ones that can be computed by a Turing machine whose run-time is bounded by a tower of exponentials of fixed height. Further constraints on functorial boxes allow to characterize the class of deterministic polytime functions, under the Curry-Howard correspondence; we get Light Linear Logic (LLL) [Gir98], together with its affine version Light Affine Logic (LAL) [AR02].

On the other side, allowing dereliction but forbidding digging, and imposing some more restrictions on the duplications, leads to Soft Linear Logic (SLL) [Laf04].

The derivations of LLL/LAL enjoy a structural invariant, called *stratification*, while for those ones in SLL the invariant will be called *boundedness*.

1.2 Stratification vs. Boundedness, a little bit more technically.

By *stratification* we summarize a logical property of LLL/LAL, which is structurally obtained by dropping dereliction and digging principles. Stratification means that the boundary of the functorial boxes, that enclose well formed sub-derivations, neither can disappear, nor can be created. So, it is natural to say that a node in a proof net of LLL/LAL, or, equivalently, a rule in a derivation of LLL/LAL, is at depth d if it is contained into d nested boxes. At the dynamic level this sums up to have that if a cut elimination step involves nodes at depth d, then only the complexity of the proof nets at depths strictly deeper than d can increase. Consequently, the control over the dimension of every single reduct becomes the control on the overall cut elimination time. By the way, the mechanism is implicit in the structural and combinatorial properties of the proof nets of LLL/LAL, and is independent from the logical soundness.

Concerning *boundedness*, recall that, in ordinary Linear Logic, $!A$ is semantically equivalent to $A^* = \bigcup_{n\in\mathbf{N}} \underbrace{(A \otimes \ldots \otimes A)}_{n}$. *Boundedness* refers to various methodologies that, informally, put $!A$ in correspondence to a *finite* subset of A^*. Computationally, this means that given a proof net, the number of copies of each box coming up during the cut elimination can somehow be statically predicted, i.e. bounded. Usually, the "deep" reason is that the interpretation of $!A$ cannot be equal to the interpretation of $!A \otimes A$ and this rules out the principle underpinning self-copying. For completeness, it is worth concluding by recalling that also Bounded Linear Logic [GSS92], of which SLL seems to catch the spirit, was conceived in accordance with the principle of boundedness just illustrated.

1.3 Goal

We are currently looking for a set of logical principles, which both Stratification and Boundedness become specific subcases of.

The intuition driving us now follows. The structural bound of the polynomial time soundness of the cut elimination for LAL can be viewed as analogous to some structural bounds that can be obtained for systems exploiting the boundedness principle. In particular, the polynomial time soundness bound of LAL can be rephrased saying that there is a constant, called *rank*, that bounds the number of copies of any sub-derivation of LAL that may be duplicated under the effect of the cut elimination.

The argument applies to SLL in the reversed direction. Specifically, the polynomial time soundness of SLL can be proved in some sense as if it was a stratified deductive system. Every level in a derivation of SLL seems to be connected to suitable sets of occurrences of the *multiplexor* rule/node that simultaneously contract many occurrences of a formula, while adding a modality in front of it.

In the coming sections we shall roughly illustrate the ideas and the conjectures that drive on going work about the two intuitions just illustrated on the polytime soundness of LAL and SLL.

$$T^{\Pi}_{\mathcal{D}}(r) = \sum_{n \text{ node of } \Pi} W^{n}_{\mathcal{D}}(r)$$

$$\begin{cases} W^{n}_{\mathcal{D}}(r) = S_{\mathcal{D}}\left(r, T^{\Pi'}_{\mathcal{D}}(R_{\mathcal{D}}(r))\right) & n \text{ root of a !-box with } \Pi' \text{ in it} \\ W^{n}_{\mathcal{D}}(r) = k_n & \text{every other } n \end{cases}$$

Fig. 1. The weight of any proof net Π of a deductive system $\mathcal{D}$, depending on a rank r. The constants k_n and the functions $R_{\mathcal{D}}, S_{\mathcal{D}}$ are specific for the deductive system $\mathcal{D}$.

$$S_{\mathcal{D}}(r, t) = r \cdot t + 1$$
$$R_{\mathcal{D}}(r) = r$$

Fig. 2. Size measure and the rank-update functions for SLL.

2 LAL as a bounded system

Saying that LAL is a bounded system means looking for a *rank*, for every proof net Π of LAL. Given any Π, the rank is a constant that only depends on Π itself. The intended use of the rank is twofold. On one side it is the constant that fixes the maximal number of copies of subnets we can produce in the course of the cut elimination at a given depth d. On the other, (a function of) the rank bounds the size of the proof nets at depth deeper than d.

Formally, the idea of *rank* is given in Figure 1. $\mathcal{D}$ is a generic deductive system, as SLL or LAL.

By letting $\mathcal{D}$ be SLL, and defining both the size measure $S_{\mathcal{D}}$ and the rank-update function $R_{\mathcal{D}}$ as in Figure 2, we get the cut elimination bound of any proof net Π of SLL described in [Laf04].

Now we try letting $\mathcal{D}$ more general. Then $T^{\Pi}_{\mathcal{D}}(r)$ is a bound on the cut elimination cost that depends on the rank r that can intuitively be bounded by the initial dimension S_{Π} of Π, namely by the number of nodes in Π. $T^{\Pi}_{\mathcal{D}}(r)$ is the sum of the weights $W^{n}_{\mathcal{D}}$ of the nodes of Π. The weight is a suitable, in fact, essentially irrelevant, constant k_n, if n is not the root of a box in LAL. Otherwise, if n is the root of a box that contains the proof net Π', the weight is a bound on a size measure $S_{\mathcal{D}}$ that depends on the rank, and on the bound on the cut elimination that can operate on Π', namely on $T^{\Pi'}_{\mathcal{D}}(v)$, for a suitable value v. In its turn, the value v must be a function $R_{\mathcal{D}}(r)$ that updates the rank for Π' which is at a deeper depth than Π.

In our case, with $\mathcal{D}$ being LAL, a possible choice for the size measure $S_{\mathcal{D}}$ and the rank-update function $R_{\mathcal{D}}$ is in Figure 3. All this should convince that LAL has a notion of rank which cannot be fixed once for all by looking at specific nodes

$$S_{\mathcal{D}}(r,t) = r \cdot t$$
$$R_{\mathcal{D}}(r) = r^2$$

Fig. 3. A possible choice for the size measure and the rank-update functions for LAL.

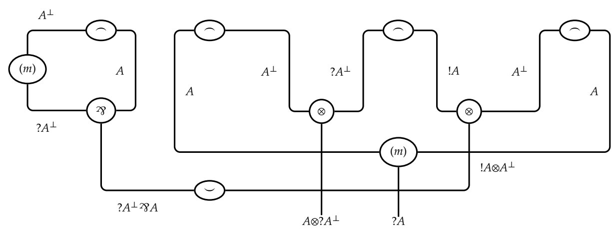

Fig. 4. The proof net Σ of SLL where we shall discover boxes.

of a given proof net, like in SLL, but which must be updated in relation to the depth we are normalizing at.

Once defined the weight, one would use it in order to give a proof of LAL polytime soundness in analogy to the proof given by Lafont in [Laf04] for SLL. In fact, the weight as defined in Figure 1 cannot be used in that way. The reason is that it does not decrease during the cut elimination. To fix this problem very likely it is necessary to extend the definition in Figure 1 with another parameter.

3 SLL as a stratified system

We give an idea about why we conjecture that the proof nets of SLL are stratified, and what we mean by stratification. SLL should be stratified in a sense weaker than LLL/LAL. We use a reasonably simple, but relevant example. Let us look at Figure 4 that contains a proof net Σ of SLL. Our idea is that every multiplexor node (m) of Σ actually corresponds to a *generalized* paragraph box, that in some way generalizes the ones we know from LLL/LAL. We call SLL^4 the hypothetical new system obtained by SLL after adding these new paragraph boxes.

SLL^4 should allow to map Σ into the proof net $\bar{\Sigma}$ in Figure 5. The leftmost multiplexor and paragraph nodes should delineate a paragraph box of which the multiplexor represents the inputs and the paragraph node is the output.

However, the rightmost multiplexor node is not so evidently connected to the remaining two paragraph nodes to form a paragraph box. Asking SLL^4 to have a more general introduction of paragraph boxes than LLL/LAL, comes in help exactly now. The introduction of paragraph boxes in SLL^4 must be so general that we must be able to prove the isomorphisms among logical operators in

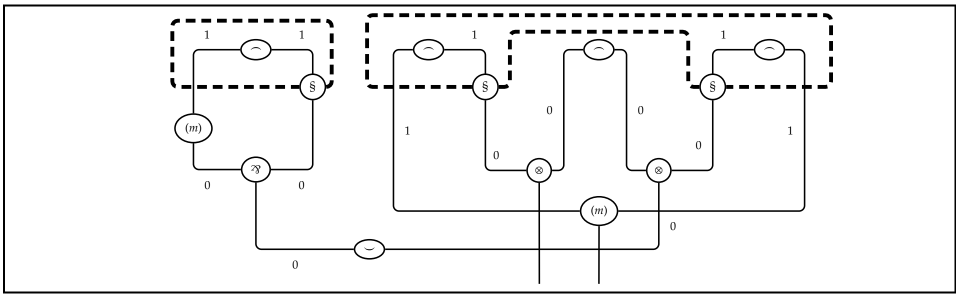

Fig. 5. The proof net $\bar{\Sigma}$ in SLL^4, image of Σ of SLL.

$$\S(A \otimes B) \simeq \S A \otimes \S B$$
$$\S(A \parr B) \simeq \S A \parr \S B$$
$$\S !A \simeq !\S A$$
$$\S\forall\alpha.A \simeq \forall\alpha.\S A$$

Fig. 6. Isomorphims among the logical operators of SLL^4.

Figure 6. Moreover, assuming that on SLL^4 the η-expansion on axioms holds, we can transform the rightmost multiplexor in Figure 5 as a node that introduces the assumptions of a paragraph box, using the two following steps:

- we η-expand the third axiom, counting from left, and
- we use the isomorphisms in Figure 6 to shift downward all the paragraphs to the right until one paragraph occurs below one of the two $\otimes$ nodes, and another paragraph node occurs below the other $\otimes$ node.

4 SLL, LLL and ML^3

The candidate system where to look for SLL^4 is ML^3 [BM08] since it enjoys η-expansion on axioms and it can prove the isomorphisms in Figure 6.

ML^3 is a proof nets system defined as the subset of LL proof nets enjoying a certain structural property called *indexing*. In [BM08], the authors show that: (i) ML^3 extends ELL, (ii) ML^3 is *weakly elementary time sound*, in the sense that every proof net may be reduced in elementary time, just following a particular reduction strategy, (iii) there exists a subsystem ML^4 of ML^3 strictly larger than LLL, and ML^4 is *weakly polynomial time sound*. One of our targets is to find an analogous system SLL^4 of ML^3 strictly containing SLL, and that will be (at least weakly) polynomial time sound too.

While searching for such a subsystem, we run into the following property.

Theorem 1 ([GRV09]). *There exists a (reasonably simple) translation @ from every proof net of* the propositional fragment of LL *into* ML^3, *that respects the cut elimination procedure.*

In particular, the restriction of @ to propositional SLL identifies a subsystem SLL^4_p of ML^3 that partially solves our research. But of course the *propositional* fragment of SLL is quite less complex than the full SLL.

5 Conclusions

Suppose for a while we shall be able to bring the definition of the embedding of SLL into a suitable SLL^4 to its end. This will not mean we shall have a system where both LAL and SLL embed, but just a system, where stratification and boundedness, used to characterize polynomial sound computations, under the Curry-Howard correspondence, actually coexist.

We conjecture it is also possible to obtain a system where both LAL and SLL embed by generalizing SLL^4, namely, by generalizing the system where we can embed SLL by means of a suitable set of modalities whose goal is to locally distinguish the origins of contracted formulæ. To this aim we plan to exploit the experience developed on Multimodal Stratified framework (MS) [RV09]. MS allows to characterize polynomial time sound systems of proof nets, with, at least in principle, an unlimited set of modal operators. So, its principles should be exploitable to add to SLL^4 the modal operators we need.

References

[AR02] A. Asperti and L. Roversi. Intuitionistic light affine logic. *ACM Transaction on Computational Logic*, 3(1):137–175, 2002.

[BM08] P. Baillot and D. Mazza. Linear logic by levels and bounded time complexity. Technical report, http://arxiv.org/abs/0801.1253v1, January 2008.

[Gir98] J.-Y. Girard. Light linear logic. *Information and Computation*, 143(2):175–204, 1998.

[GRV09] M. Gaboardi, L. Roversi, and L. Vercelli. Multiplicative Exponential Linear Logic can be levelled. http://www.di.unito.it/˜vercelli/works/soli-part-I.pdf, Submitted, 2009.

[GSS92] J.-Y. Girard, A. Scedrov, and P. Scott. Bounded linear logic: A modular approach to polynomial time computability. *Theoretical Computer Science*, 97:1–66, 1992.

[Laf04] Y. Lafont. Soft linear logic and polynomial time. *Theoretical Computer Science*, 318:163–180, 2004.

[RV09] L. Roversi and L. Vercelli. Some Complexity and Expressiveness results on Multimodal and Stratified Proof-nets. In *TYPES*, 2009. http://www.di.unito.it/˜vercelli/works/ms-part-I.pdf.