Interface in Linear Logic : A Finite System of Generators

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

Abstract

Linear Logic [Gir87] is a logic that has two presentations: a classical sequent calculus and a more original presentation using proof nets. Proof nets are graphs of connectors and boxes connected together by *links*. Each cell has a given numbers of *ports* and links connect ports two by two (no sharing). Boxes serve to bound nets.

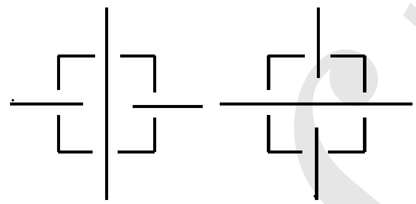
This formalism was introduced because several proofs in sequent calculus which are not essentially different (see for example the exchange rule or commutable rules) correspond to the same proof net. Nets go beyond logic because they may be used to perform β-reduction in λ-calculus [Lam90, Mac95] or used as a model of calculus as interaction nets [Laf90].

However, not every net corresponds to linear logic proofs and a criterion serves to distinguish between bad *proof structures* and *proof nets*. Using linear logic with mix rule (saying that two proofs side by side form a proof), this criterion looks if the graph have not *bad cycle*. The same mechanism may be used in interaction nets to ensure that their reduction will not create *dead-lock* (a very bad configuration where every cell is blocked forever by an other one in a cycle). To check if a net is correct, a set of switching positions is associated to each king of cell. A net is said to be correct if for every switching of the cells, no cycle appears.

To characterize the connection property of a bit of net (called a module [DR89]) with other nets, its *interface* may be computed. Now, two modules can be connected together if there interfaces are *orthogonal*. More precisely, an interface is a set of *partitions* and a partition is an equivalence relation between the external ports of a module. Two partitions are said to be orthogonal if the composition creates no cycle passing alternatively by one partition and the other one. For instance the partitions ab/cd and $a/bc/d$ are orthogonal but ab/cd and ad/bc are not (the cycle $ab|bc|cd|da$). Two interfaces are orthogonal when partitions of the two interfaces are orthogonal two by two.

Since interfaces are set over partitions, one can ask if every set of partition correspond to a net. We prove that the answer is no with with the classical connectives of linear logic. However, if we add a special combinator called *crossroad* to this system, we can prove that this element and the usual connectors ($\otimes$ and $\wp$) of linear logic are generators of every set of partitions.

The crossroad has 4 ports and two switching:



However, this result can only be apply to linear logic with mix rule. Without this rule, we have prove that there is no finite set of cells that generates all interfaces (this notion is a little more complex for those proof nets).

References

[DR89] V. Danos and L. Regnier. The structure of multiplicatives. *Archive for Mathematical Logic*, 28:181–203, 1989.

[Gir87] J.-Y. Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[Laf90] Y. Lafont. Interaction nets. In *Proc. 17-th ACM Symp. on Principles of Programming Languages, San Francisco*, pages 95–108, January 1990.

[Lam90] J. Lamping. An algorithm for optimal lambda calculus reduction. In *Seventeenth Annual ACM Symposium on Principles of Programming Languages*, pages 16–30. ACM, January 1990.

[Mac95] Ian Mackie. A λ-evaluator based on interaction nets. In Chris Hankin, Ian Mackie, and Raja Nagarajan, editors, *Theory and Formal Methods 1994: Proceedings of the Second Imperial College, Department of Computing, Workshop on Theory and Formal Methods, Møller Center, Cambridge, UK, 11–14 September 1994.* Imperial College Press, 1995.