Linear Logic with Polarities

Olivier LAURENT

1 Polarities in Linear Logic

In [4], J.-Y. Girard defines the system LC for classical logic and its denotational semantics based on a notion of *polarized formulas*, distinguishing two dual classes of formulas, *positive* and *negative* formulas:

$$\begin{array}{lclcccccccccccccc} P & ::= & X^\perp & \mid & P \otimes P & \mid & P \oplus P & \mid & \exists X P & \mid & 1 & \mid & 0 & \mid & !N \\ N & ::= & X & \mid & N \parr N & \mid & N \,\&\, N & \mid & \forall X N & \mid & \perp & \mid & \top & \mid & ?P \end{array}$$

From a syntactical point of view, negativity corresponds to the *reversibility* of logical rules whereas positivity corresponds to the *focalization* property. Furthermore negative formulas make structural rules admissible:

$$\frac{\vdash \Gamma, N, N}{\vdash \Gamma, N} \qquad \frac{\vdash \Gamma}{\vdash \Gamma, N} \qquad \frac{\vdash \mathcal{N}, N}{\vdash \mathcal{N}, !N}$$

where N is any negative formula and $\mathcal{N}$ is any context of negative formulas.

We define LLP to be the fragment of Linear Logic dealing only with polarized formulas, with these generalized structural rules (and an additional technical constraint on the $\top$-rule).

LLP has proved to be a very good system for encoding classical logic, with a simple notion of proof-nets decomposing the reduction of Parigot's $\lambda\mu$-calculus [8] (just as proof-nets decompose β-reduction) and a good semantics. The relation between LLP and linear logic is similar to the relation between classical and intuitionistic logic, thus giving a linear account for CPS translations.

2 Polarized proof-nets

The restriction to polarized formulas allows the definition of a very simple *correctness criterion* for LLP proof-nets (with additives, quantifiers and constants) by adding an orientation [7]:

- positive ↔ upwards
- negative ↔ downwards

and checking an acyclicity-connexity condition on the resulting oriented graph.

In these proof-nets any positive node has an implicit box structure similar to the explicit !-box associated to !-nodes. This property allows to define a cut elimination procedure which in particular solves the commutative cut problem of proof-nets for full linear logic.

3 The $\lambda\mu$-calculus

The traditional translations of classical logic into LL, based either on $\mathsf{LK}^{\mathsf{tq}}$ [3] or on CPS translations, make intensive use of exponential rules. Typically the t-translation is based on the encoding of the classical implication: $A \to B \rightsquigarrow\; !?A \multimap ?B$. Thanks to the generalization of structural rules, LLP allows a very simple encoding of classical logic, based on the same translation of the classical implication as the translation of the intuitionistic one in Linear Logic: $A \to B \rightsquigarrow\; !A \multimap B$.

In this way we get an encoding of $\lambda\mu$-calculus into polarized proof-nets [6] which simulates normalization by cut-elimination. This translation may easily be adapted to Peter Selinger's extension of $\lambda\mu$-calculus with pairing and a disjunctive construction [10], yielding a surjective reduction preserving mapping from Selinger's system into LLP proof-nets. As a consequence, Selinger's control categories give a categorical model of LLP.

The study of the inverse of the translation mapping leads to the definition of an equivalence relation on $\lambda\mu$-terms, extending the σ-equivalence on λ-terms [9]. Interestingly, the σ-equivalence for $\lambda\mu$-calculus also identifies normal terms. As LLP proof-nets realize the quotient we immediately get that two equivalent terms are indistinguishable syntactically as well as semantically.

4 Game semantics (work in progress)

We define a game semantics for LLP, much in the spirit of AJM games [1], by introducing a notion of *polarized games* based on the correspondence::

- negative ↔ opponent starts;
- positive ↔ player starts.

Multiplicative connectives are defined à la Lamarche [2]. In the spirit of Girard's ludics [5] there are two shifting operators that add a fake move to a game thus just changing its polarity.

In this framework, AJM's "intuitionistic games" appear like the negative counterpart of the logical constructions of linear logic. Typically the AJM tensor product $\odot$ is a negative version of the linear tensor product $\otimes$ related to the latter using a shifting operator: $\downarrow (A \odot B) \simeq \downarrow A \otimes \downarrow B$.

References

[1] Samson Abramsky, Radha Jagadeesan, and Pasquale Malacaria. Full abstraction for PCF (extended abstract). In Masami Hagiya and John C. Mitchell, editors, *Theoretical Aspects of Computer Software. International Symposium TACS'94*, volume 789 of *Lecture Notes in Computer Science*, pages 1–15. Springer, April 1994.

[2] François Lamarche. Sequentiality, games and linear logic (announcement). In *Workshop on Categorical Logic in Computer Science*. Publications of the Computer Science Department of Aarhus University, DAIMI PB-397-II, 1992.

[3] Vincent Danos, Jean-Baptiste Joinet, and Harold Schellinx. A new deconstructive logic: linear logic. *Journal of Symbolic Logic*, 62(3):755–807, September 1997.

[4] Jean-Yves Girard. A new constructive logic: classical logic. *Mathematical Structures in Computer Science*, 1(3):255–296, 1991.

[5] Jean-Yves Girard. Locus solum. Private communication, March 2000.

[6] Olivier Laurent. Polarized proof-nets and $\lambda\mu$-calculus. Submitted. Available at `ftp://iml.univ-mrs.fr/pub/olaurent/lmpn.ps.gz`, December 1999.

[7] Olivier Laurent. Polarized proof-nets: proof-nets for LC (extended abstract). In Jean-Yves Girard, editor, *Typed Lambda Calculi and Applications '99*, volume 1581 of *Lecture Notes in Computer Science*, pages 213–227. Springer, April 1999.

[8] Michel Parigot. $\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction. In *Proceedings of Internationnal Conference on Logic Programming and Automated Deduction*, volume 624 of *Lecture Notes in Computer Science*, pages 190–201. Springer, 1992.

[9] Laurent Regnier. Une équivalence sur les lambda-termes. *Theoretical Computer Science*, 126:281–292, 1994.

[10] Peter Selinger. Control categories and duality: on the categorical semantics of the lambda-mu calculus. Available at `ftp://ftp.math.lsa.umich.edu/pub/selinger/control.ps.gz`, To appear in *Mathematical Structures in Computer Science*, 2000.