An exercise on Light Affine Logic

November 24, 2006

Notations: The lambda-calculus application will be denoted $(t)u$. We will write $\lambda x_1 x_2.t$ for $\lambda x_1.\lambda x_2.t$, and $(t)\ u_1 \dots\ u_n$ for $(\dots((t)\ u_1)\dots u_n)$.

We recall that the type of Church integers in F (resp. in LAL, *Light Affine Logic*) is $N = \forall\alpha.(\alpha \to \alpha) \to (\alpha \to \alpha)$ (resp. $N^{LAL} = \forall\alpha.!(\alpha \to \alpha) \to \S(\alpha \to \alpha)$). The Church integer for n is written $\underline{n}$.

Exercise:

We consider the lamda-terms $t = \lambda msx.(s)(s)(m)sx$ and $u = \lambda n.(n)\ t\ \underline{0}$. A type derivation for these two terms in intuitionnistic second-order logic sequent-calculus (system F) is given below (the first few rules of the two derivations, in particular the subderivation of $\vdash \underline{0} : N$, have been omitted for simplification).

Using these two derivations, give derivations in LAL for t and u with respectively conclusion type $N^{LAL} \multimap N^{LAL}$ and $N^{LAL} \to \S N^{LAL}$.

What do these two terms compute? Could we have given type $N^{LAL} \to N^{LAL}$ to term u?

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{y : \alpha \to \alpha, s_1 : \alpha \to \alpha, s_2 : \alpha \to \alpha, x : \alpha \ \vdash\ (s_1)(s_2)(y)x : \alpha}{y : \alpha \to \alpha, s_1 : \alpha \to \alpha, s_2 : \alpha \to \alpha \ \vdash\ \lambda x.(s_1)(s_2)(y)x : \alpha \to \alpha} \qquad s_3 : \alpha \to \alpha \ \vdash\ s_3 : \alpha \to \alpha}{m : (\alpha \to \alpha) \to (\alpha \to \alpha), s_1 : \alpha \to \alpha, s_2 : \alpha \to \alpha, s_3 : \alpha \to \alpha \ \vdash\ \lambda x.(s_1)(s_2)(m)s_3x : \alpha \to \alpha}}{m : N, s_1 : \alpha \to \alpha, s_2 : \alpha \to \alpha, s_3 : \alpha \to \alpha \ \vdash\ \lambda x.(s_1)(s_2)(m)s_3x : \alpha \to \alpha}}{m : N, s : \alpha \to \alpha \ \vdash\ \lambda x.(s)(s)(m)sx : \alpha \to \alpha}}{m : N \ \vdash\ \lambda sx.(s)(s)(m)sx : (\alpha \to \alpha) \to (\alpha \to \alpha)}}{m : N \ \vdash\ \lambda sx.(s)(s)(m)sx : N}}{\vdash\ \lambda msx.(s)(s)(m)sx : N \to N}
$$

$$
\dfrac{\dfrac{\dfrac{\dfrac{\vdash\ \underline{0} : N \qquad z : N \ \vdash\ z : N}{y : N \to N \ \vdash\ (y)\underline{0} : N} \qquad \vdash\ t : N \to N}{n : (N \to N) \to (N \to N) \ \vdash\ (n)\ t\ \underline{0} : N}}{n : N \ \vdash\ (n)\ t\ \underline{0} : N}}{\vdash\ \lambda n.(n)\ t\ \underline{0} : N \to N}
$$

CORRECTION

We basically need to add ! and § rules in the derivations in such a way that:

- the contraction rules are correct (that is to say applied to formulas of the form $!A$),
- the conclusion of the derivation is the expected one.

Two suitable decorations of the derivations with !, § rules are given below. Note that in the first derivation: a ! or § rule must be applied before the contraction on the variables s_i, so as to give s_1 and s_2 type $!(\alpha \multimap \alpha)$; moreover it can only be a § rule, since there is more than one formula on the l.h.s. of the sequent.

$$
\begin{prooftree}
\AxiomC{$y : \alpha \multimap \alpha, s_1 : \alpha \multimap \alpha, s_2 : \alpha \multimap \alpha, x : \alpha \vdash (s_1)(s_2)(y)x : \alpha$}
\UnaryInfC{$y : \alpha \multimap \alpha, s_1 : \alpha \multimap \alpha, s_2 : \alpha \multimap \alpha, x : \alpha \vdash \lambda x.(s_1)(s_2)(y)x : \alpha \multimap \alpha$}
\UnaryInfC{$y : \S(\alpha \multimap \alpha), s_1 : !(\alpha \multimap \alpha), s_2 : !(\alpha \multimap \alpha) \vdash \lambda x.(s_1)(s_2)(y)x : \S(\alpha \multimap \alpha)$}
\AxiomC{$s_3 : !(\alpha \multimap \alpha) \vdash s_3 : !(\alpha \multimap \alpha)$}
\BinaryInfC{$m : !(\alpha \multimap \alpha) \multimap \S(\alpha \multimap \alpha), s_1 : !(\alpha \multimap \alpha), s_2 : !(\alpha \multimap \alpha), s_3 : !(\alpha \multimap \alpha) \vdash \lambda x.(s_1)(s_2)(m)s_3x : \S(\alpha \multimap \alpha)$}
\UnaryInfC{$m : N^{LAL}, s_1 : !(\alpha \multimap \alpha), s_2 : !(\alpha \multimap \alpha), s_3 : !(\alpha \multimap \alpha) \vdash \lambda x.(s_1)(s_2)(m)s_3x : \S(\alpha \multimap \alpha)$}
\UnaryInfC{$m : N^{LAL}, s : !(\alpha \multimap \alpha) \vdash \lambda x.(s)(s)(m)sx : \S(\alpha \multimap \alpha)$}
\UnaryInfC{$m : N^{LAL} \vdash \lambda sx.(s)(s)(m)sx : !(\alpha \multimap \alpha) \multimap \S(\alpha \multimap \alpha)$}
\UnaryInfC{$m : N^{LAL} \vdash \lambda sx.(s)(s)(m)sx : N^{LAL}$}
\UnaryInfC{$\vdash \lambda msx.(s)(s)(m)sx : N^{LAL} \multimap N^{LAL}$}
\end{prooftree}
$$

$$
\begin{prooftree}
\AxiomC{$\vdash \underline{0} : N^{LAL}$}
\AxiomC{$z : N^{LAL} \vdash z : N^{LAL}$}
\BinaryInfC{$y : (N^{LAL} \multimap N^{LAL}) \vdash (y)\underline{0} : N^{LAL}$}
\UnaryInfC{$y : \S(N^{LAL} \multimap N^{LAL}) \vdash (y)\underline{0} : \S N^{LAL}$}
\AxiomC{$\vdash t : (N^{LAL} \multimap N^{LAL})$}
\UnaryInfC{$\vdash t : !(N^{LAL} \multimap N^{LAL})$}
\BinaryInfC{$n : !(N^{LAL} \multimap N^{LAL}) \multimap \S(N^{LAL} \multimap N^{LAL}) \vdash (n)\ t\ \underline{0} : \S N^{LAL}$}
\UnaryInfC{$n : N^{LAL} \vdash (n)\ t\ \underline{0} : \S N^{LAL}$}
\UnaryInfC{$\vdash \lambda n.(n)\ t\ \underline{0} : N^{LAL} \multimap \S N^{LAL}$}
\end{prooftree}
$$

The term t computes the function adding 2 to an integer. The term u, given an integer n, iterates n times t starting from 0, hence it computes the *doubling* function on Church integers.

Note that if we could type term t with type $N^{LAL} \multimap N^{LAL}$, then it could be iterated just as term u. However the iteration of the doubling function would give the exponentiation function on unary integers, which is not PTIME. Hence such a typing for t is not possible because it would contradict the PTIME soundness result for LAL.