Implicit computational complexity :
Light types for lambda-calculus

Patrick Baillot

CNRS-LIPN, Université Paris 13

patrick.baillot@lipn.univ-paris13.fr

Lecture (S. Martini) at Bertinoro Intenational Spring School in CS

march 10th 2006

< > - +

Light Affine Logic (LAL)

Light linear logic and its affine variant were introduced as logics with polynomial time *dynamics*, t.i.t.s. normalization/cut-elimination.

Language of LAL formulas

$$A, B ::= \alpha \mid A \multimap B \mid A \otimes B \mid !A \mid \S A \mid \forall \alpha . A$$

- LAL *captures* Ptime: All and only polynomial time functions can be represented by LAL proofs of conclusion $\vdash \mathbb{B} \multimap \S^k \mathbb{B}$, for $k \in \mathbb{N}$.
- however: proofs have to be normalized (evaluated) using *proof-nets*.

thus:

underlying intuition	:	lambda-calculus
$\neq$ actual machinery needed	:	proof-nets

< > - +

Skimming LAL . . .

In practice, we rarely use types of the following forms

$$A \multimap !B, \quad \S!A \multimap B, \quad !A\ .$$

However ! is definitely needed for types of the kind:

$$!A \multimap B$$

No surprise there: LL precisely arose from this decomposition of the intuitionistic arrow . . .

Let us thus consider the fragment of LAL where ! is only used in types $!A \multimap B$:
to make this clearer, let us even drop ! and consider instead a non-linear arrow $\Rightarrow$, together with $\multimap$.

< > - +

Dual Light Affine Logic (DLAL)

the resulting system is called *Dual Light Affine Logic*, as it will handle two kinds of variables (linear *vs* non-linear ones).
language of DLAL types:

$$A, B ::= \alpha \mid A \multimap B \mid A \Rightarrow B \mid \S A \mid \forall \alpha . A$$

translation $(.)^* : DLAL \longrightarrow LAL$:

- $(A \Rightarrow B)^* = !A^* \multimap B^*$,
- $(.)^*$ commutes to other connectives.

Note that the following types are not in the image of DLAL :

$$A \multimap !B, \quad \S!A \multimap B, \quad !A \; .$$

< > - +

DLAL typing judgements

mixed judgements: the context is split into two parts

$$\Gamma;\Delta \vdash t : C$$

linear $x : A \in \Delta$ at most 1 occurrence

harmless

non-linear $y : B \in \Gamma$ any number of occurrence

dangerous: *beware of substitutions*

a variable can change status from linear to non-linear (with some conditions)

< > - +

DLAL, sequent calculus style

$$\frac{}{;x : A \vdash x : A}\ \text{(Id)}$$

$$\frac{\Gamma_1;\Delta_1 \vdash u:A \quad \Gamma_2; x:A, \Delta_2 \vdash t:C}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2 \vdash t[u/x]:C}\ (Cut)$$

$$\frac{\Gamma_1;\Delta_1, x : A \vdash t : B}{\Gamma_1;\Delta_1 \vdash \lambda x.t : A \multimap B}\ (\multimap \text{r})$$

$$\frac{\Gamma_1;\Delta_1 \vdash u:A \quad \Gamma_2; x:B, \Delta_2 \vdash t:C}{\Gamma_1,\Gamma_2; y:A \multimap B,\Delta_1,\Delta_2 \vdash t[(y\ u)/x]:C}\ (\multimap l)$$

$$\frac{\Gamma_1, x : A;\Delta_1 \vdash t : B}{\Gamma_1;\Delta_1 \vdash \lambda x.t : A \Rightarrow B}\ (\Rightarrow \text{r})$$

$$\frac{;z:D \vdash u:A \quad \Gamma; x:B, \Delta \vdash t:C}{z:D,\Gamma; y:A \Rightarrow B,\Delta \vdash t[(y\ u)/x]:C}\ (\Rightarrow l)(*)$$

$$\frac{\Gamma_1;\Delta_1 \vdash t : A}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2 \vdash t : A}\ \text{(Weak)}$$

$$\frac{x_1 : A, x_2 : A, \Gamma_1;\Delta_1 \vdash t : B}{x : A, \Gamma_1;\Delta_1 \vdash t[x/x_1, x/x_2] : B}\ \text{(Cntr)}$$

$$\frac{;\Gamma, x_1 : B_1, \ldots, x_n : B_n \vdash t : A}{\Gamma; x_1 : \S B_1, \ldots, x_n : \S B_n \vdash t : \S A}\ (\S\ )$$

$$\frac{\Gamma; x:A[B/\alpha],\Delta \vdash t:C}{\Gamma; x:\forall\alpha.A,\Delta \vdash t:C}\ (\forall l)$$

$$\frac{\Gamma;\Delta \vdash t:A}{\Gamma;\Delta \vdash t:\forall\alpha.A}\ (\forall r),\ \alpha \text{ not free in } \Gamma,\Delta$$

(*) $z : D$ can be absent.

DLAL, sequent calculus style

$$\frac{}{;x : A \vdash x : A} \text{(Id)}$$

$$\frac{\Gamma_1; \Delta_1 \vdash u : A \quad \Gamma_2; x : A, \Delta_2 \vdash t : C}{\Gamma_1, \Gamma_2; \Delta_1, \Delta_2 \vdash t[u/x] : C} (Cut)$$

$$\frac{\Gamma_1; \Delta_1, x : A \vdash t : B}{\Gamma_1; \Delta_1 \vdash \lambda x.t : A \multimap B} (\multimap \text{r})$$

$$\frac{\Gamma_1; \Delta_1 \vdash u : A \quad \Gamma_2; x : B, \Delta_2 \vdash t : C}{\Gamma_1, \Gamma_2; y : A \multimap B, \Delta_1, \Delta_2 \vdash t[(y\ u)/x] : C} (\multimap l)$$

$$\frac{\Gamma_1, x : A; \Delta_1 \vdash t : B}{\Gamma_1; \Delta_1 \vdash \lambda x.t : A \Rightarrow B} (\Rightarrow \text{r})$$

$$\frac{;z : D \vdash u : A \quad \Gamma; x : B, \Delta \vdash t : C}{z : D, \Gamma; y : A \Rightarrow B, \Delta \vdash t[(y\ u)/x] : C} (\Rightarrow l)(*)$$

$$\frac{\Gamma_1; \Delta_1 \vdash t : A}{\Gamma_1, \Gamma_2; \Delta_1, \Delta_2 \vdash t : A} \text{(Weak)}$$

$$\frac{x_1 : A, x_2 : A, \Gamma_1; \Delta_1 \vdash t : B}{x : A, \Gamma_1; \Delta_1 \vdash t[x/x_1, x/x_2] : B} \text{(Cntr)}$$

$$\frac{;\Gamma, x_1 : B_1, \ldots, x_n : B_n \vdash t : A}{\Gamma; x_1 : \S B_1, \ldots, x_n : \S B_n \vdash t : \S A} (\S\ )$$

$$\frac{\Gamma; x : A[B/\alpha], \Delta \vdash t : C}{\Gamma; x : \forall \alpha.A, \Delta \vdash t : C} (\forall l)$$

$$\frac{\Gamma; \Delta \vdash t : A}{\Gamma; \Delta \vdash t : \forall \alpha.A} (\forall r), \alpha \text{ not free in } \Gamma, \Delta$$

(*) $z : D$ can be absent.

< > - +

DLAL, sequent calculus style

$$\frac{}{;x:A \vdash x:A}\ \text{(Id)}$$

$$\frac{\Gamma_1;\Delta_1 \vdash u:A \quad \Gamma_2;x:A,\Delta_2 \vdash t:C}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2 \vdash t[u/x]:C}\ (Cut)$$

$$\frac{\Gamma_1;\Delta_1, x:A \vdash t:B}{\Gamma_1;\Delta_1 \vdash \lambda x.t : A \multimap B}\ (\multimap \text{r})$$

$$\frac{\Gamma_1;\Delta_1 \vdash u:A \quad \Gamma_2;x:B,\Delta_2 \vdash t:C}{\Gamma_1,\Gamma_2;y:A \multimap B,\Delta_1,\Delta_2 \vdash t[(y\ u)/x]:C}\ (\multimap l)$$

$$\frac{\Gamma_1, x:A;\Delta_1 \vdash t:B}{\Gamma_1;\Delta_1 \vdash \lambda x.t : A \Rightarrow B}\ (\Rightarrow \text{r})$$

$$\frac{;z:D \vdash u:A \quad \Gamma;x:B,\Delta \vdash t:C}{z:D,\Gamma;y:A \Rightarrow B,\Delta \vdash t[(y\ u)/x]:C}\ (\Rightarrow l)(*)$$

$$\frac{\Gamma_1;\Delta_1 \vdash t:A}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2 \vdash t:A}\ \text{(Weak)}$$

$$\frac{x_1:A, x_2:A,\Gamma_1;\Delta_1 \vdash t:B}{x:A,\Gamma_1;\Delta_1 \vdash t[x/x_1,x/x_2]:B}\ \text{(Cntr)}$$

$$\frac{;\Gamma, x_1:B_1,\dots,x_n:B_n \vdash t:A}{\Gamma;x_1:\S B_1,\dots,x_n:\S B_n \vdash t:\S A}\ (\S\ )$$

$$\frac{\Gamma;x:A[B/\alpha],\Delta \vdash t:C}{\Gamma;x:\forall\alpha.A,\Delta \vdash t:C}\ (\forall l)$$

$$\frac{\Gamma;\Delta \vdash t:A}{\Gamma;\Delta \vdash t:\forall\alpha.A}\ (\forall r),\ \alpha \text{ not free in } \Gamma,\Delta$$

(*) $z:D$ can be absent.

< > - +

DLAL, sequent calculus style

$$\frac{}{;x : A \vdash x : A}\ \text{(Id)}$$

$$\frac{\Gamma_1;\Delta_1 \vdash u:A \quad \Gamma_2;x:A,\Delta_2 \vdash t:C}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2 \vdash t[u/x]:C}\ (Cut)$$

$$\frac{\Gamma_1;\Delta_1, x : A \vdash t : B}{\Gamma_1;\Delta_1 \vdash \lambda x.t : A \multimap B}\ (\multimap \text{r})$$

$$\frac{\Gamma_1;\Delta_1 \vdash u:A \quad \Gamma_2;x:B,\Delta_2 \vdash t:C}{\Gamma_1,\Gamma_2;y:A \multimap B,\Delta_1,\Delta_2 \vdash t[(y\ u)/x]:C}\ (\multimap l)$$

$$\frac{\Gamma_1, x : A;\Delta_1 \vdash t : B}{\Gamma_1;\Delta_1 \vdash \lambda x.t : A \Rightarrow B}\ (\Rightarrow \text{r})$$

$$\frac{;z:D \vdash u:A \quad \Gamma;x:B,\Delta \vdash t:C}{z:D,\Gamma;y:A\Rightarrow B,\Delta \vdash t[(y\ u)/x]:C}\ (\Rightarrow l)(*)$$

$$\frac{\Gamma_1;\Delta_1 \vdash t : A}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2 \vdash t : A}\ \text{(Weak)}$$

$$\frac{x_1 : A, x_2 : A, \Gamma_1;\Delta_1 \vdash t : B}{x : A, \Gamma_1;\Delta_1 \vdash t[x/x_1, x/x_2] : B}\ \text{(Cntr)}$$

$$\frac{;\Gamma, x_1 : B_1, \ldots, x_n : B_n \vdash t : A}{\Gamma; x_1 : \S B_1, \ldots, x_n : \S B_n \vdash t : \S A}\ (\S\ )$$

$$\frac{\Gamma;x:A[B/\alpha],\Delta \vdash t:C}{\Gamma;x:\forall\alpha.A,\Delta \vdash t:C}\ (\forall l)$$

$$\frac{\Gamma;\Delta \vdash t:A}{\Gamma;\Delta \vdash t:\forall\alpha.A}\ (\forall r),\ \alpha \text{ not free in } \Gamma, \Delta$$

(*) $z : D$ can be absent.

DLAL, sequent calculus style

$$\frac{}{;x : A \vdash x : A}\ \text{(Id)}$$

$$\frac{\Gamma_1;\Delta_1 \vdash u:A \quad \Gamma_2; x:A, \Delta_2 \vdash t:C}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2 \vdash t[u/x]:C}\ (Cut)$$

$$\frac{\Gamma_1;\Delta_1, x : A \vdash t : B}{\Gamma_1;\Delta_1 \vdash \lambda x.t : A \multimap B}\ (\multimap \text{r})$$

$$\frac{\Gamma_1;\Delta_1 \vdash u:A \quad \Gamma_2; x:B, \Delta_2 \vdash t:C}{\Gamma_1,\Gamma_2; y:A \multimap B, \Delta_1,\Delta_2 \vdash t[(y\ u)/x]:C}\ (\multimap l)$$

$$\frac{\Gamma_1, x : A;\Delta_1 \vdash t : B}{\Gamma_1;\Delta_1 \vdash \lambda x.t : A \Rightarrow B}\ (\Rightarrow \text{r})$$

$$\frac{; z:D \vdash u:A \quad \Gamma; x:B, \Delta \vdash t:C}{z:D, \Gamma; y:A \Rightarrow B, \Delta \vdash t[(y\ u)/x]:C}\ (\Rightarrow l)(*)$$

$$\frac{\Gamma_1;\Delta_1 \vdash t : A}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2 \vdash t : A}\ \text{(Weak)}$$

$$\frac{x_1 : A, x_2 : A, \Gamma_1;\Delta_1 \vdash t : B}{x : A, \Gamma_1;\Delta_1 \vdash t[x/x_1, x/x_2] : B}\ \text{(Cntr)}$$

$$\frac{;\Gamma, x_1 : B_1, \dots, x_n : B_n \vdash t : A}{\Gamma; x_1 : \S B_1, \dots, x_n : \S B_n \vdash t : \S A}\ (\S\ )$$

$$\frac{\Gamma; x:A[B/\alpha], \Delta \vdash t:C}{\Gamma; x:\forall\alpha.A, \Delta \vdash t:C}\ (\forall l)$$

$$\frac{\Gamma;\Delta \vdash t:A}{\Gamma;\Delta \vdash t:\forall\alpha.A}\ (\forall r),\ \alpha \text{ not free in } \Gamma,\Delta$$

(*) $z : D$ can be absent.

< > - +

DLAL, sequent calculus style

$$\frac{}{;x:A\vdash x:A}\ \text{(Id)}$$

$$\frac{\Gamma_1;\Delta_1\vdash u:A \quad \Gamma_2;x:A,\Delta_2\vdash t:C}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2\vdash t[u/x]:C}\ (Cut)$$

$$\frac{\Gamma_1;\Delta_1,x:A\vdash t:B}{\Gamma_1;\Delta_1\vdash \lambda x.t:A\multimap B}\ (\multimap \text{r})$$

$$\frac{\Gamma_1;\Delta_1\vdash u:A \quad \Gamma_2;x:B,\Delta_2\vdash t:C}{\Gamma_1,\Gamma_2;y:A\multimap B,\Delta_1,\Delta_2\vdash t[(y\ u)/x]:C}\ (\multimap l$$

$$\frac{\Gamma_1,x:A;\Delta_1\vdash t:B}{\Gamma_1;\Delta_1\vdash \lambda x.t:A\Rightarrow B}\ (\Rightarrow \text{r})$$

$$\frac{;z:D\vdash u:A \quad \Gamma;x:B,\Delta\vdash t:C}{z:D,\Gamma;y:A\Rightarrow B,\Delta\vdash t[(y\ u)/x]:C}\ (\Rightarrow l)(*)$$

$$\frac{\Gamma_1;\Delta_1\vdash t:A}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2\vdash t:A}\ \text{(Weak)}$$

$$\frac{x_1:A,x_2:A,\Gamma_1;\Delta_1\vdash t:B}{x:A,\Gamma_1;\Delta_1\vdash t[x/x_1,x/x_2]:B}\ \text{(Cntr)}$$

$$\frac{;\Gamma,x_1:B_1,\ldots,x_n:B_n\vdash t:A}{\Gamma;x_1:\S B_1,\ldots,x_n:\S B_n\vdash t:\S A}\ (\S)$$

$$\frac{\Gamma;x:A[B/\alpha],\Delta\vdash t:C}{\Gamma;x:\forall\alpha.A,\Delta\vdash t:C}\ (\forall l)$$

$$\frac{\Gamma;\Delta\vdash t:A}{\Gamma;\Delta\vdash t:\forall\alpha.A}\ (\forall r),\ \alpha \text{ not free in } \Gamma,\Delta$$

(*) $z:D$ can be absent.

Proposition 0 *If $\mathcal{D}$ has conclusion $\Gamma;\Delta\vdash_{DLAL} t:A$ then $|t|\leq|\mathcal{D}|$.*

< > - +

DLAL, sequent calculus style

$$\frac{}{;x:A\vdash x:A}\ \text{(Id)}$$

$$\frac{\Gamma_1;\Delta_1\vdash u:A \quad \Gamma_2;x:A,\Delta_2\vdash t:C}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2\vdash t[u/x]:C}\ (Cut)$$

$$\frac{\Gamma_1;\Delta_1,x:A\vdash t:B}{\Gamma_1;\Delta_1\vdash \lambda x.t:A\multimap B}\ (\multimap \text{r})$$

$$\frac{\Gamma_1;\Delta_1\vdash u:A \quad \Gamma_2;x:B,\Delta_2\vdash t:C}{\Gamma_1,\Gamma_2;y:A\multimap B,\Delta_1,\Delta_2\vdash t[(y\ u)/x]:C}\ (\multimap l)$$

$$\frac{\Gamma_1,x:A;\Delta_1\vdash t:B}{\Gamma_1;\Delta_1\vdash \lambda x.t:A\Rightarrow B}\ (\Rightarrow \text{r})$$

$$\frac{;z:D\vdash u:A \quad \Gamma;x:B,\Delta\vdash t:C}{z:D,\Gamma;y:A\Rightarrow B,\Delta\vdash t[(y\ u)/x]:C}\ (\Rightarrow l)(*)$$

$$\frac{\Gamma_1;\Delta_1\vdash t:A}{\Gamma_1,\Gamma_2;\Delta_1,\Delta_2\vdash t:A}\ \text{(Weak)}$$

$$\frac{x_1:A,x_2:A,\Gamma_1;\Delta_1\vdash t:B}{x:A,\Gamma_1;\Delta_1\vdash t[x/x_1,x/x_2]:B}\ \text{(Cntr)}$$

$$\frac{;\Gamma,x_1:B_1,\ldots,x_n:B_n\vdash t:A}{\Gamma;x_1:\S B_1,\ldots,x_n:\S B_n\vdash t:\S A}\ (\S)$$

$$\frac{\Gamma;x:A[B/\alpha],\Delta\vdash t:C}{\Gamma;x:\forall\alpha.A,\Delta\vdash t:C}\ (\forall l)$$

$$\frac{\Gamma;\Delta\vdash t:A}{\Gamma;\Delta\vdash t:\forall\alpha.A}\ (\forall r),\ \alpha \text{ not free in } \Gamma,\Delta$$

(*) $z:D$ can be absent.

< > - +

Type derivations in DLAL

Proposition 0 *if* $\Gamma; \Delta \vdash_{DLAL} t : A$ *and* $x \in \Delta$*, then* x *has at most one occurrence in* t *(*x *linear).*

Proposition 0 *If* $\Gamma; \Delta \vdash_{DLAL} t : A$ *then* $!\Gamma^*, \Delta^* \vdash_{LAL} t : A^*$*.*
We obtain in this way a simulation from DLAL to LAL.

The *depth* of a DLAL derivation is the depth of the corresponding LAL derivation.

< > - +

DLAL data types

- unary integers

LAL: N^{LAL}

$$\forall\alpha.!(\alpha \multimap \alpha) \multimap \S(\alpha \multimap \alpha),$$

DLAL N^{DLAL}

$$\forall\alpha.(\alpha \multimap \alpha) \Rightarrow \S(\alpha \multimap \alpha)\ .$$

- binary lists

LAL: $\mathbb{B}^{LAL}$

$$\forall\alpha.!(\alpha \multimap \alpha) \multimap !(\alpha \multimap \alpha) \multimap \S(\alpha \multimap \alpha),$$

DLAL $\mathbb{B}^{DLAL}$

$$\forall\alpha.(\alpha \multimap \alpha) \Rightarrow (\alpha \multimap \alpha) \Rightarrow \S(\alpha \multimap \alpha)\ .$$

< > - +

Types in DLAL

examples of types:

$$\vdash addition : \quad N \multimap N \multimap N$$
$$\vdash double : \quad N \Rightarrow \S N$$

The functions representable in DLAL are given by types $\mathbb{B} \multimap \S^k\mathbb{B}$.

Typing of iterators:

$$iter_A = \lambda fxn.\ (n\ f\ x)\ : (A \multimap A) \Rightarrow \S A \multimap N \multimap \S A$$

$$iter^{\mathbb{B}}{}_A = \lambda f_1 f_2 xw.\ (w\ f_1\ f_2\ x)\ : (A \multimap A) \Rightarrow (A \multimap A) \Rightarrow \S A \multimap \mathbb{B} \multimap \S A$$

< > - +

Coercions in DLAL

Type coercions on data-types are needed to compose programs suitably.

in LAL: $coer^{p,q} : N \multimap \S^{p+1} !^q N$

in DLAL, the following rules are derivable:

$$\frac{n : N; \Delta \vdash t : A}{; n : N, \S\Delta \vdash C_1[t] : \S A} \text{ (coerc1)} \qquad \frac{\Gamma; n : \S N, \Delta \vdash t : A}{\Gamma; n : N, \Delta \vdash C_2[t] : A} \text{ (coerc2)}$$

with C_i contexts such that $C_i[t]$ is extensionally equivalent to t (represents the same function)

Example	$mult$	:	$N \Rightarrow (N \multimap \S N)$	
	$mult'$	:	$N \multimap \S(N \multimap \S N)$	by (coerc1)
	$mult''$	:	$N \multimap N \multimap \S\S N$	by (coerc2)

Proposition 1 *If $P \in \mathbb{N}[X]$, then there exists a term t_P representing P and an integer k such that: $\vdash_{DLAL} t_P : N \multimap \S^k N$.*

DLAL: complexity bounds

Proposition 2 *DLAL satisfies the subject-reduction property.*

Theorem 3 (strong polynomial bound) *If t is typeable in DLAL with a derivation of depth d, then any sequence of β-reducts of t has length bounded by $O(|t|^{2^d})$.*
Moreover this reduction is performed in time $O(|t|^{2^{d+2}})$ on a Turing machine.

Remarks:

- we are dealing here with β-reduction, and not anymore with proof-net reduction;
- this bound holds for any reduction strategy;
- in particular, if $\vdash t : \mathbb{B} \multimap \S^k \mathbb{B}$ then we can normalize $(t\ \underline{w})$ in a number of steps (and time) polynomial in $length(w)$.

Theorem 4 *The functions representable by terms typeable in DLAL are exactly the functions computable in polynomial time.*

Conclusion

- LL brings a logical, proof-based approach to implicit computational complexity,
- this is one approach relating
  ICC on the one hand
  and functional programming, typing, proofs on the other . . .
- however, the *intensional* expressivity is limited :
  all polynomial time *functions* are representable, but some common polynomial time algorithms are not (directly) typeable
  $\longrightarrow$ more flexible systems in the typing approach ?