Linear and Dependent Types, Part 1-2

Neel Krishnaswami
University of Cambridge

July 6, 2017

1 Papers

- *Linear logic, monads, and the lambda calculus*. Nick Benton, Philip Wadler. 11th IEEE Symposium on Logic in Computer Science (LICS). New Brunswick, New Jersey, July 1996.
- *L^3 : A Linear Language with Locations*. Amal Ahmed, Matthew Fluet, Greg Morrisett. Fundamenta Informaticae XXI (2001). 1001 – 1053.
- *Integrating Linear and Dependent Types*. Neelakantan R. Krishnaswami, Pierre Pradic, Nick Benton. ACM SIGPLAN Symposium on Principles of Programming Programming Languages (POPL). Mumbai, India. January 2015.

2 Propositional Adjoint Logic

2.1 Syntax

$$
\begin{array}{llcl}
\text{Intuitionistic Types} & X & ::= & 1 \mid X \times Y \mid X \to Y \mid \mathsf{G}\,A \\
\text{Linear Types} & A & ::= & \mathsf{I} \mid A \otimes B \mid A \multimap B \mid \mathsf{F}\,X \\
\\
\text{Intuitionistic Contexts} & \Gamma & ::= & \cdot \mid \Gamma, x : X \\
\text{Linear Contexts} & \Delta & ::= & \cdot \mid \Delta, a : A \\
\\
\text{Intuitionistic Terms} & e & ::= & () \mid (e, e') \mid \mathsf{fst}\,e \mid \mathsf{snd}\,e \mid \lambda x.\,e \mid e\,e' \mid \mathsf{G}(t) \mid x \\
\text{Linear Terms} & t & ::= & () \mid \mathsf{let}\,() = t\,\mathsf{in}\,t' \mid (t, t') \mid \mathsf{let}\,(x, y) = t\,\mathsf{in}\,t' \\
 & & \mid & \lambda a.\,t \mid t\,t' \mid \mathsf{F}(e) \mid \mathsf{let}\,\mathsf{F}(x) = t\,\mathsf{in}\,t' \mid \mathsf{run}(e) \mid a
\end{array}
$$

2.2 Rules

$$\boxed{\Gamma \vdash e : X}$$

$$\frac{}{\Gamma \vdash () : 1}\ 1\mathrm{I} \qquad\qquad \text{(no 1E)}$$

$$\frac{\Gamma \vdash e : X \qquad \Gamma \vdash e' : Y}{\Gamma \vdash (e, e') : X \times Y}\ \times\mathrm{I} \qquad \frac{\Gamma \vdash e : X \times Y}{\Gamma \vdash \mathsf{fst}(e) : X}\ \times\mathrm{E}_1 \qquad \frac{\Gamma \vdash e : X \times Y}{\Gamma \vdash \mathsf{snd}(e) : Y}\ \times\mathrm{E}_2$$

$$\frac{\Gamma, x : X \vdash e : Y}{\Gamma \vdash \lambda x.\, e : X \to Y}\ \to\mathrm{I} \qquad \frac{\Gamma \vdash e : X \to Y \qquad \Gamma \vdash e' : X}{\Gamma \vdash e\ e' : Y}\ \to\mathrm{E}$$

$$\frac{\Gamma; \cdot \vdash t : A}{\Gamma \vdash \mathsf{G}(t) : \mathsf{G}(A)}\ \mathsf{GI} \qquad \frac{x : X \in \Gamma}{\Gamma \vdash x : X}\ \textsc{VarI}$$

$$\boxed{\Gamma; \Delta \vdash t : A}$$

$$\frac{}{\Gamma; \cdot \vdash () : \mathsf{I}}\ \mathsf{II} \qquad \frac{\Gamma; \Delta \vdash t : \mathsf{I} \qquad \Gamma; \Delta' \vdash t' : C}{\Gamma; \Delta, \Delta' \vdash \mathsf{let}\ () = t\ \mathsf{in}\ t' : C}\ \mathsf{IE}$$

$$\frac{\Gamma; \Delta \vdash t : A \qquad \Gamma; \Delta' \vdash t' : B}{\Gamma; \Delta, \Delta' \vdash (t, t') : A \otimes B}\ \otimes\mathrm{I} \qquad \frac{\Gamma; \Delta \vdash t : A \otimes B \qquad \Gamma; \Delta', a : A, b : B \vdash t' : C}{\Gamma; \Delta, \Delta' \vdash \mathsf{let}\ (a, b) = t\ \mathsf{in}\ t' : C}\ \otimes\mathrm{E}$$

$$\frac{\Gamma; \Delta, a : A \vdash t : B}{\Gamma; \Delta \vdash \lambda a.\, t : A \multimap B}\ \multimap\mathrm{I} \qquad \frac{\Gamma; \Delta \vdash t : A \multimap B \qquad \Gamma; \Delta' \vdash t' : A}{\Gamma; \Delta, \Delta' \vdash t\ t' : B}\ \multimap\mathrm{E}$$

$$\frac{\Gamma \vdash e : X}{\Gamma; \cdot \vdash \mathsf{F}(e) : \mathsf{F}(X)}\ \mathsf{FI} \qquad \frac{\Gamma; \Delta \vdash t : \mathsf{F}(X) \qquad \Gamma, x : X; \Delta' \vdash t' : C}{\Gamma; \Delta, \Delta' \vdash \mathsf{let}\ \mathsf{F}(x) = t\ \mathsf{in}\ t' : C}\ \mathsf{FE}$$

$$\frac{\Gamma \vdash e : \mathsf{G}(A)}{\Gamma; \cdot \vdash \mathsf{run}(e) : A}\ \mathsf{GE} \qquad \frac{}{\Gamma; a : A \vdash a : A}\ \textsc{VarL}$$

2.3 Substitution Properties

Lemma 1. *(Weakening) We have that:*

- *If* $\Gamma \vdash e : Y$ *then* $\Gamma, x : X \vdash e : Y$.
- *If* $\Gamma; \Delta \vdash t : A$ *then* $\Gamma, x : X; \Delta \vdash t : A$.

Theorem 1. *(Substitution) We have that:*

- *If* $\Gamma \vdash e : X$ *and* $\Gamma, x : X \vdash e' : Y$ *then* $\Gamma \vdash [e/x]e' : Y$.
- *If* $\Gamma \vdash e : X$ *and* $\Gamma, x : X; \Delta \vdash t : A$ *then* $\Gamma; \Delta \vdash [e/x]t : A$.
- *If* $\Gamma; \Delta \vdash t : A$ *and* $\Gamma; \Delta', a : A \vdash t' : B$ *then* $\Gamma \vdash [t/a]t' : B$.

3 State

$$
\begin{array}{llcl}
\text{Intuitionistic Types} & X & ::= & \ldots \mid \mathsf{Ptr}(x) \mid \forall x.\, X \mid \exists x.\, X \\
\text{Linear Types} & A & ::= & \ldots \mid \mathsf{Cap}(x, X) \mid \mathsf{T}(A) \mid \forall x.\, A \mid \exists x.\, X \\
\\
\text{Intuitionistic Contexts} & \Gamma & ::= & \cdot \mid \Gamma, x : X \\
\text{Linear Contexts} & \Delta & ::= & \cdot \mid \Delta, a : A \\
\text{Location Contexts} & \Sigma & ::= & \cdot \mid \Sigma, i \\
\\
\text{Intuitionistic Terms} & e & ::= & \ldots \mid l \\
\text{Linear Terms} & t & ::= & \ldots \mid * \mid \mathsf{val}(t) \mid \mathsf{let}\ \mathsf{val}(x) = t\ \mathsf{in}\ t' \\
 & & \mid & \mathsf{let}\ (i, x, c) = \mathsf{new}\ \mathsf{in}\ t \mid \mathsf{free}(e, t); t' \\
 & & \mid & \mathsf{let}\ (x, c) = \mathsf{read}(e, t)\ \mathsf{in}\ t' \mid \mathsf{let}\ a = e :=_t e'\ \mathsf{in}\ t' \\
\\
\text{Stores} & \sigma & ::= & \cdot \mid \sigma, l : v
\end{array}
$$

3.1 Operational Semantics

$\boxed{e \mapsto e'}$ $\boxed{t \mapsto t'}$

$$(\lambda x.\, e)\ v \mapsto [v/x]e$$

$$\mathsf{fst}\,(v, v') \mapsto v$$

$$\mathsf{snd}\,(v, v') \mapsto v'$$

$$\frac{e \mapsto e'}{I[e] \mapsto I[e']}$$

$$(\lambda a.\, t)\ u \mapsto [u/a]t$$

$$\mathsf{let}\ () = ()\ \mathsf{in}\ t \mapsto t$$

$$\mathsf{let}\ (a, b) = (u, u')\ \mathsf{in}\ t \mapsto [u/a, u'/b]t$$

$$\mathsf{run}(\mathsf{G}(t)) \mapsto t$$

$$\mathsf{let}\ \mathsf{F}(x) = \mathsf{F}(v)\ \mathsf{in}\ t \mapsto [v/x]t$$

$$\frac{t \mapsto t'}{L_t[e] \mapsto L_t[e']}$$

$$\frac{e \mapsto e'}{L_e[e] \mapsto L_t[e']}$$

$$\boxed{\langle \sigma; t\rangle \mapsto \langle \sigma'; t'\rangle}$$

$$\langle \sigma; \mathsf{let}\ \mathsf{val}(a) = \mathsf{val}(u)\ \mathsf{in}\ t\rangle \mapsto \langle \sigma; [u/a]t\rangle$$

$$\langle \sigma; \mathsf{let}\ (i, x, c) = \mathsf{new}\ \mathsf{in}\ t\rangle \mapsto \langle [\sigma \mid l : ()]; [l/i, l/x, */c]t\rangle \qquad \text{with } l \notin \mathrm{dom}(\sigma)$$

$$\langle [\sigma \mid l : v]; \mathsf{free}(l, *); t\rangle \mapsto \langle \sigma; t\rangle$$

$$\langle [\sigma \mid l : v]; \mathsf{let}\ (x, c) = \mathsf{read}(l, *)\ \mathsf{in}\ t\rangle \mapsto \langle [\sigma \mid l : v]; [v/x, */c]t\rangle$$

$$\langle [\sigma \mid l : -]; \mathsf{let}\ a = l :=_* v\ \mathsf{in}\ t\rangle \mapsto \langle [\sigma \mid l : v]; [*/a]t\rangle$$

$$\frac{\langle \sigma; t\rangle \mapsto \langle \sigma'; t'\rangle}{\langle \sigma; K[t]\rangle \mapsto \langle \sigma'; K[t']\rangle}$$

3.2 Typing

$$\frac{\Gamma; \Delta \vdash_\Sigma t : A}{\Gamma; \Delta \vdash_\Sigma \mathsf{val}(t) : \mathsf{T}(A)} \qquad \frac{\Gamma; \Delta \vdash_\Sigma t : \mathsf{T}(A) \qquad \Gamma; \Delta', a : A \vdash_\Sigma \Delta' : t'\mathsf{T}(C)}{\Gamma; \Delta, \Delta' \vdash_\Sigma \mathsf{let}\ \mathsf{val}(a) = t\ \mathsf{in}\ t' : \mathsf{T}(C)}$$

$$\frac{\Gamma, x : \mathsf{Ptr}(i); \Delta, a : \mathsf{Cap}(x, 1) \vdash_{\Sigma, i} t : \mathsf{T}(C)}{\Gamma; \Delta \vdash_\Sigma \mathsf{let}\ (i, x, a) = \mathsf{new}\ \mathsf{in}\ t : \mathsf{T}(C)} \qquad \frac{\Gamma \vdash_\Sigma e : \mathsf{Ptr}(i) \qquad \Gamma; \Delta \vdash_\Sigma t : \mathsf{Cap}(i, X) \qquad \Gamma; \Delta' \vdash_\Sigma t' : \mathsf{T}(C)}{\Gamma; \Delta, \Delta' \vdash_\Sigma \mathsf{free}(e, t); t' : \mathsf{T}(C)}$$

$$\frac{\Gamma \vdash_\Sigma e : \mathsf{Ptr}(i) \qquad \Gamma; \Delta \vdash_\Sigma t : \mathsf{Cap}(i, X) \qquad \Gamma, x : X; \Delta', a : \mathsf{Cap}(i, X) \vdash_\Sigma t' : \mathsf{T}(C)}{\Gamma; \Delta, \Delta' \vdash_\Sigma \mathsf{let}\ (x, a) = \mathsf{read}(e, t)\ \mathsf{in}\ t' : \mathsf{T}(C)}$$

$$\frac{\Gamma \vdash_\Sigma e : \mathsf{Ptr}(i) \qquad \Gamma; \Delta \vdash_\Sigma t : \mathsf{Cap}(i, X) \qquad \Gamma \vdash_\Sigma e' : Y \qquad \Gamma; \Delta', a : \mathsf{Cap}(i, Y) \vdash_\Sigma t' : \mathsf{T}(C)}{\Gamma; \Delta, \Delta' \vdash_\Sigma \mathsf{let}\ a = e :=_t e'\ \mathsf{in}\ t' : \mathsf{T}(C)}$$

3.3 Logical Relation

We write $\sigma \# \sigma'$ to mean that the domains of σ and σ' are disjoint. We write $\sigma \cdot \sigma'$ to mean the concatenation of σ and σ'. This is a partial operation defined only when their domains are disjoint. We write $e \mapsto^* e'$ to mean the transitive closure of evaluation.

$$
\begin{array}{lcl}
[\![1]\!]\ \rho & = & \{((),())\} \\
[\![X \times Y]\!]\ \rho & = & \{(v, v') \mid (\mathsf{fst}(v), \mathsf{fst}(v')) \in \mathcal{E}[\![X]\!]\rho \wedge (\mathsf{snd}(v), \mathsf{snd}(v')) \in \mathcal{E}[\![Y]\!]\rho\} \\
[\![X \to Y]\!]\ \rho & = & \{(v, v') \mid \forall e.\ \text{if } (e, e') \in \mathcal{E}[\![X]\!]\rho \text{ then } (v\ e, v'\ e') \in \mathcal{E}[\![Y]\!]\rho\} \\
[\![\mathsf{Ptr}(x)]\!]\rho & = & \{(\rho(x), \rho(x))\} \\
[\![\mathsf{G}(A)]\!]\rho & = & \{(\mathsf{G}(t), \mathsf{G}(t')) \mid (\langle \cdot; t\rangle, \langle \cdot; t'\rangle) \in \mathcal{E}[\![A]\!]\rho\} \\
 & & \\
\mathcal{E}[\![X]\!]\rho & = & \{(e, e') \mid \exists v, v'.\ e \mapsto^* v \wedge e' \mapsto^* v' \wedge (v, v') \in [\![X]\!]\rho\}
\end{array}
$$

$$
\begin{array}{lcl}
[\![I]\!]\ \rho & = & \{(\langle \cdot; ()\rangle, \langle \cdot; ()\rangle)\} \\
[\![A \otimes B]\!]\ \rho & = & \left\{ (\langle \sigma_A \cdot \sigma_B; (u_A, u_B)\rangle, \langle \sigma'_A \cdot \sigma'_B; (u'_A, u'_B)\rangle) \;\middle|\; \begin{array}{l} (\langle \sigma_A; u_A\rangle, \langle \sigma'_A; u'_A\rangle) \in [\![A]\!]\rho \wedge \\ (\langle \sigma_B; u_B\rangle, \langle \sigma'_B; u'_B\rangle) \in [\![B]\!]\rho \end{array} \right\} \\
[\![A \multimap B]\!]\ \rho & = & \left\{ (\langle \sigma; u\rangle, \langle \sigma'; u'\rangle) \;\middle|\; \begin{array}{l} \forall t_A, \sigma_A, t'_A, \sigma'_A \text{ such that } \sigma \# \sigma_A \text{ and } \sigma' \# \sigma'_A. \\ \text{if } (\langle \sigma_A; t_A\rangle, \langle \sigma'_A; t'_A\rangle) \in \mathcal{L}[\![A]\!] \\ \text{then } (\langle \sigma \cdot \sigma_A; u\ t_A\rangle, \langle \sigma' \cdot \sigma'_A; u'\ t'_A\rangle) \in \mathcal{L}[\![B]\!] \end{array} \right\} \\
[\![\mathsf{Cap}(x, X)]\!]\rho & = & \{(\langle [l : v]; *\rangle, \langle [l : v']; *\rangle) \mid l = \rho(x) \wedge (v, v') \in [\![X]\!]\rho\} \\
 & & \\
[\![\mathsf{F}(X)]\!]\rho & = & \{(\langle \cdot; \mathsf{F}(v)\rangle, \langle \cdot; \mathsf{F}(v')\rangle) \mid (v, v') \in [\![X]\!]\rho\} \\
 & & \\
[\![\mathsf{T}(A)]\!]\rho & = & \left\{ (\langle \sigma_1; u_1\rangle, \langle \sigma'_1; u'_1\rangle) \;\middle|\; \begin{array}{l} \forall \phi, \phi' \text{ such that } \phi \# \sigma_1 \text{ and } \phi' \# \sigma'_1. \\ \exists \sigma_2, u_2, \sigma'_2, u'_2. \\ \qquad \langle \phi \cdot \sigma_1; u_1\rangle \mapsto^* \langle \phi \cdot \sigma_2; \mathsf{val}(u_2)\rangle \wedge \\ \qquad \langle \phi' \cdot \sigma'_1; u'_1\rangle \mapsto^* \langle \phi' \cdot \sigma'_2; \mathsf{val}(u'_2)\rangle \wedge \\ \qquad (\langle \sigma_2; u_2\rangle, \langle \sigma'_2; u'_2\rangle) \in [\![A]\!]\rho \end{array} \right\} \\
 & & \\
\mathcal{L}[\![A]\!]\rho & = & \left\{ (\langle \sigma; t\rangle, \langle \sigma'; t'\rangle) \;\middle|\; \begin{array}{l} \exists u, u'. t \mapsto^* u \wedge \\ \qquad t \mapsto^* u \wedge \\ \qquad (\langle \sigma; t\rangle, \langle \sigma'; t'\rangle) \in [\![A]\!]\rho \end{array} \right\}
\end{array}
$$