Proof nets, Optimal Reduction, Complexity Bounded Logics

Paolo Coppola

Università di Udine

Summary

- Type Inference in EAL (Coppola, Martini. 2001)
- Principal typing in EAL (Copppola, Ronchi della Rocca. 2003)
- Phase semantics for EAL (Dal Lago, Martini. 2003)

Elementary Affine Logic (EAL)

- Unrestricted weakening

$$\frac{\Gamma \vdash B}{\Gamma, A \vdash B}\ (Weak.)$$

- No δ and ϵ rules. There is a unique exponential rule

$$\frac{A_1, \ldots, A_n \vdash B}{!A_1, \ldots, !A_n \vdash !B}\ (!)$$

Notice that $\vdash !A \multimap A$ is not provable in EAL

Type inference in EAL

Why are we interested in EAL-type inference?

- In (Asperti, Coppola, Martini. 2000) we showed EAL-typed λ-terms powerful enough to encode elementary TM computations, but typing was “ad hoc”

Type inference in EAL

Why are we interested in EAL-type inference?

- In (Asperti, Coppola, Martini. 2000) we showed EAL-typed λ-terms powerful enough to encode elementary TM computations, but typing was “ad hoc”
- EAL-typeable terms are reducible within the abstract Lamping’s Optimal Reduction Algorithm.

Type inference in EAL

Why are we interested in EAL-type inference?

- In (Asperti, Coppola, Martini. 2000) we showed EAL-typed λ-terms powerful enough to encode elementary TM computations, but typing was “ad hoc”
- EAL-typeable terms are reducible within the abstract Lamping’s Optimal Reduction Algorithm.

  $\Rightarrow$ no computational overhead due to the oracle

Type inference in EAL

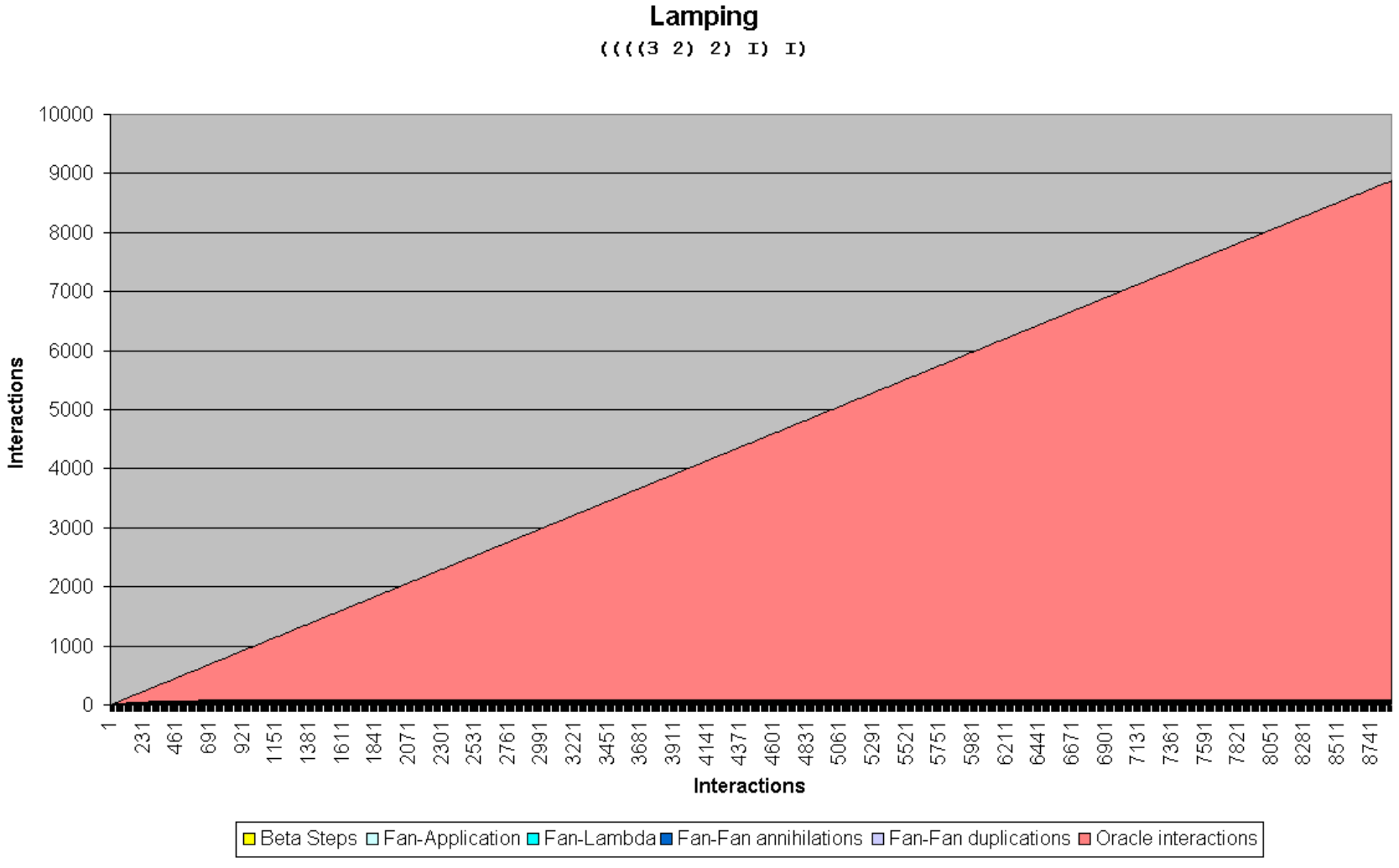
Why are we interested in EAL-type inference?

- In (Asperti, Coppola, Martini. 2000) we showed EAL-typed λ-terms powerful enough to encode elementary TM computations, but typing was “ad hoc”
- EAL-typeable terms are reducible within the abstract Lamping’s Optimal Reduction Algorithm.

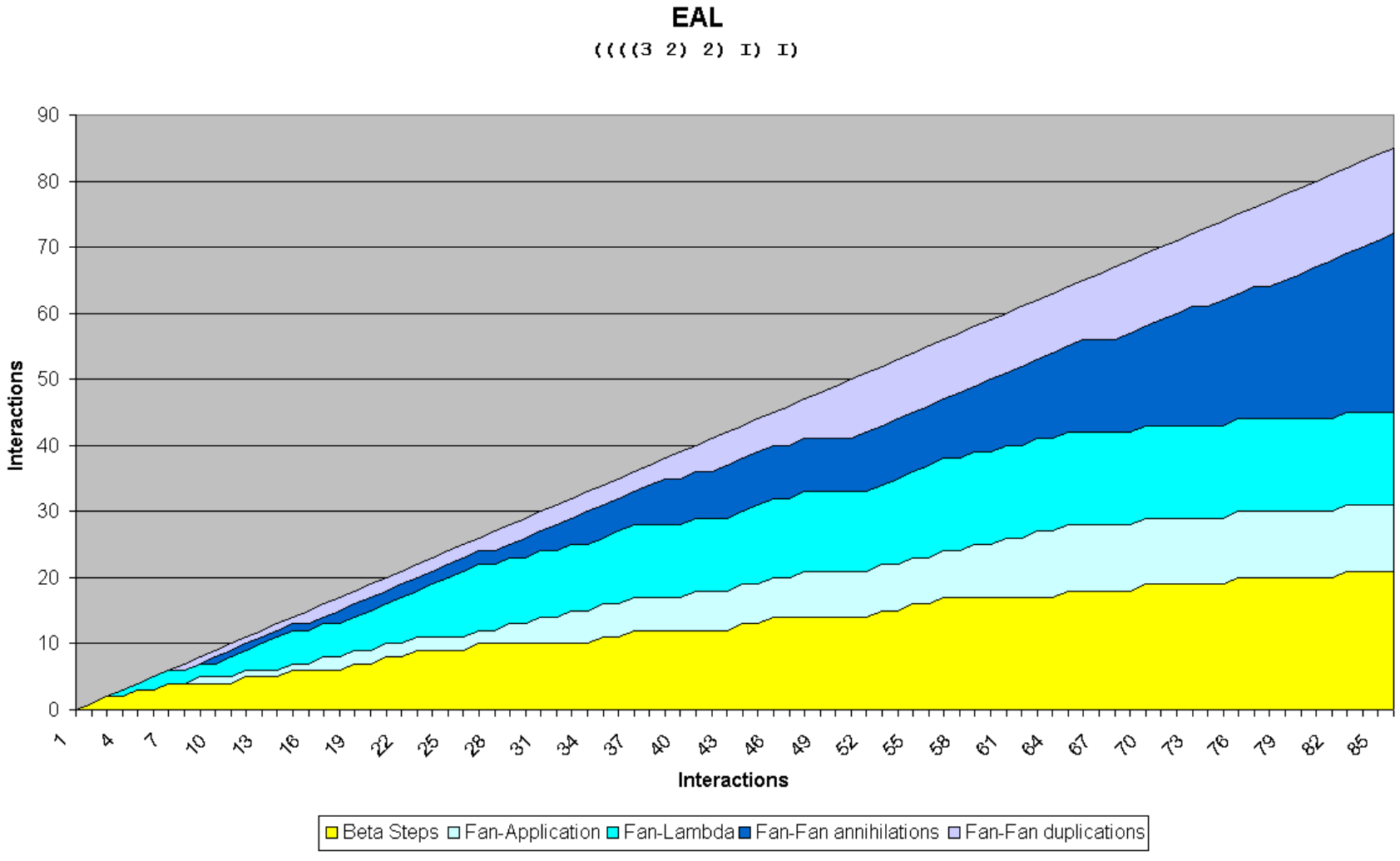
  $\Rightarrow$ no computational overhead due to the oracle

  $\Rightarrow$ dramatic improvement of performances

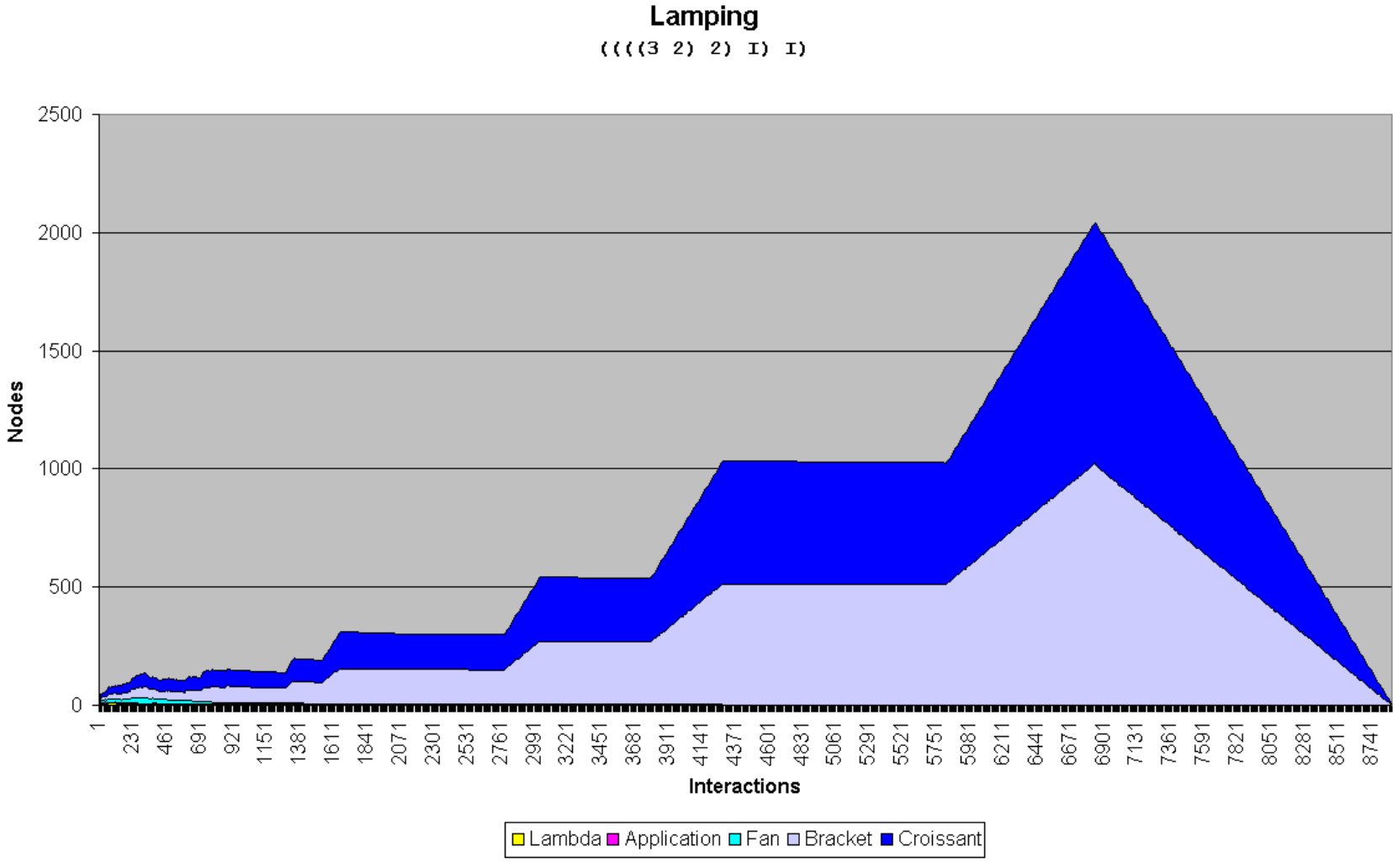
Improvement of performances



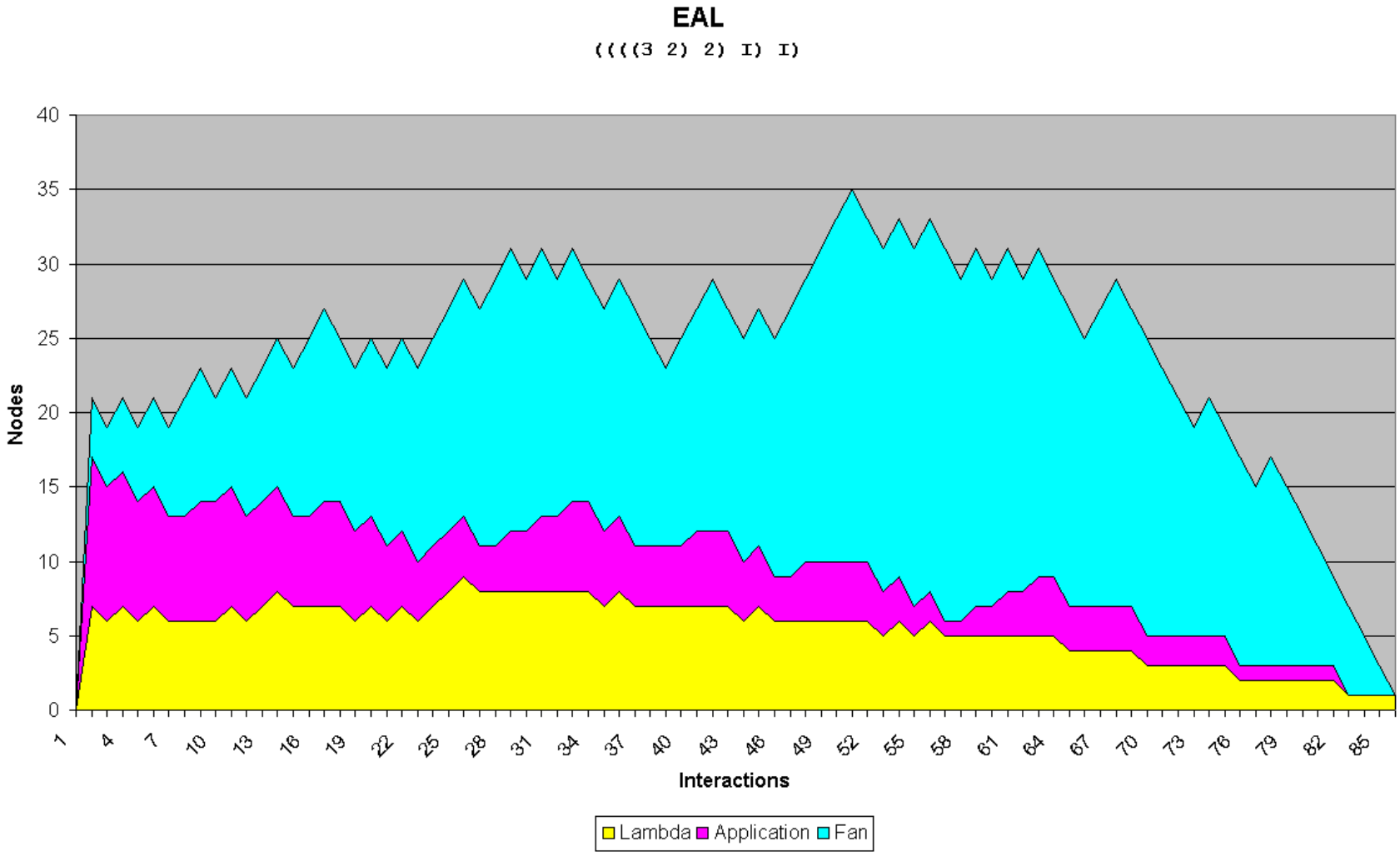
Improvement of performances



Improvement of performances

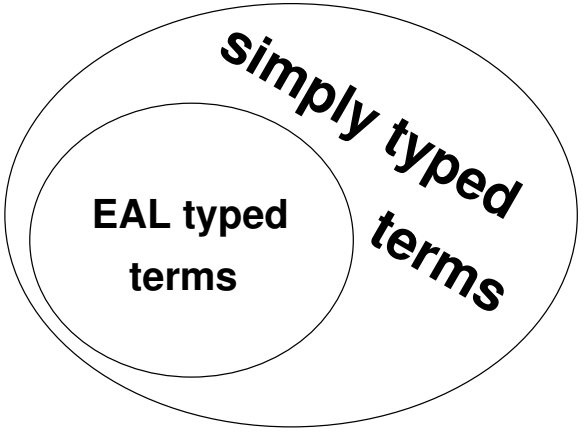


Improvement of performances



Decorations

- Any lambda term with an EAL type has also a simple type.
- There are simply typed terms not typeable in EAL



Given a simple typed lambda term, we have to:

1. find MAXIMAL DECORATIONS
2. solving set of LINEAR CONSTRAINTS

Maximal decorations

- EAL rules are Intuitionistic Logic plus !-introduction rule.
- The idea is to interleave n_i !-introduction rules with the "intuitionistic" rules of the simple type derivation.

Example

Consider the following simple type derivation:

$$\dfrac{\dfrac{\dfrac{w : o \vdash w : o \quad y : o \vdash y : o}{x : o \to o, y : o \vdash (x\ y) : o} \quad z : o \vdash z : o}{x : o \to o, x : o \to o, y : o \vdash (x(x\ y)) : o}}{x : o \to o, y : o \vdash (x(x\ y)) : o}$$

Interleaving n_i !-introduction rules it becomes...

Example

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{w{:}o \vdash w{:}o}{w{:}!^{n_1}o \vdash w{:}!^{n_1}o}\,!^{n_1} \quad \dfrac{y{:}o \vdash y{:}o}{y{:}!^{n_2}o \vdash y{:}!^{n_2}o}\,!^{n_2}}{x{:}!^{n_2}o \multimap !^{n_1}o, y{:}!^{n_2}o \vdash (x\ y){:}!^{n_1}o}\multimap L}{x{:}!^{n_3}(!^{n_2}o \multimap !^{n_1}o), y{:}!^{n_2+n_3}o \vdash (x\ y){:}!^{n_1+n_3}o}\,!^{n_3} \quad \dfrac{z{:}o \vdash z{:}o}{z{:}!^{n_4}o \vdash z{:}!^{n_4}o}\,!^{n_4}}{x{:}!^{n_3}(!^{n_2}o \multimap !^{n_1}o), x{:}!^{n_1+n_3}o \multimap !^{n_4}o, y{:}!^{n_2+n_3}o \vdash (x(x\ y)){:}!^{n_4}o}\multimap L}{x{:}!^{n_3+n_5}(!^{n_2}o \multimap !^{n_1}o), x{:}!^{n_5}(!^{n_1+n_3}o \multimap !^{n_4}o), y{:}!^{n_2+n_3+n_5}o \vdash (x(x\ y)){:}!^{n_4+n_5}o}\,!^{n_5}$$

$$\dfrac{x{:}!^{n_3+n_5}(!^{n_2}o \multimap !^{n_1}o), x{:}!^{n_5}(!^{n_1+n_3}o \multimap !^{n_4}o), y{:}!^{n_2+n_3+n_5}o \vdash (x(x\ y)){:}!^{n_4+n_5}o}{x{:}!^{n_3+n_5}(!^{n_2}o \multimap !^{n_1}o), y{:}!^{n_2+n_3+n_5}o \vdash (x(x\ y)){:}!^{n_4+n_5}o}\ contr?$$

Example

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{w{:}o \vdash w{:}o}{w{:}!^{n_1}o \vdash w{:}!^{n_1}o}\,!^{n_1} \quad \dfrac{y{:}o \vdash y{:}o}{y{:}!^{n_2}o \vdash y{:}!^{n_2}o}\,!^{n_2}}{x{:}!^{n_2}o \multimap !^{n_1}o, y{:}!^{n_2}o \vdash (x\ y){:}!^{n_1}o}\multimap L}{x{:}!^{n_3}(!^{n_2}o \multimap !^{n_1}o), y{:}!^{n_2+n_3}o \vdash (x\ y){:}!^{n_1+n_3}o}\,!^{n_3} \quad \dfrac{z{:}o \vdash z{:}o}{z{:}!^{n_4}o \vdash z{:}!^{n_4}o}\,!^{n_4}}{x{:}!^{n_3}(!^{n_2}o \multimap !^{n_1}o), x{:}!^{n_1+n_3}o \multimap !^{n_4}o, y{:}!^{n_2+n_3}o \vdash (x(x\ y)){:}!^{n_4}o}\multimap L}{x{:}!^{n_3+n_5}(!^{n_2}o \multimap !^{n_1}o), x{:}!^{n_5}(!^{n_1+n_3}o \multimap !^{n_4}o), y{:}!^{n_2+n_3+n_5}o \vdash (x(x\ y)){:}!^{n_4+n_5}o}\,!^{n_5}}{x{:}!^{n_3+n_5}(!^{n_2}o \multimap !^{n_1}o), y{:}!^{n_2+n_3+n_5}o \vdash (x(x\ y)){:}!^{n_4+n_5}o}\,contr?$$

- Every EAL-derivation is obtained instantiating $n1, \ldots, n_5$
- Contraction is almost correct: we need to impose some constraints...

Linear Constraints

$$\frac{x :!^{n_3+n_5}\left(!^{n_2}o \multimap !^{n_1}o\right), x :!^{n_5}\left(!^{n_1+n_3}o \multimap !^{n_4}o\right) \cdots \vdash \cdots}{x :!^{n_3+n_5}\left(!^{n_2}o \multimap !^{n_1}o\right) \cdots \vdash \cdots}\ contr?$$

1. Unification:

$$!^{n_3+n_5}\left(!^{n_2}o \multimap !^{n_1}o\right) =!^{n_5}\left(!^{n_1+n_3}o \multimap !^{n_4}o\right)$$

$$\begin{cases} n_3+n_5=n_5 \\ n_2=n_1+n_3 \\ n_1=n_4 \end{cases} \Leftrightarrow \begin{cases} n_3=0 \\ n_1=n_2=n4 \end{cases}$$

Simple type structure already unified $\Rightarrow$ no substitution, just linear constraints

Linear Constraints - 2

$$\begin{cases} n_3 = 0 \\ n_1 = n_2 = n4 \end{cases}$$

$$\frac{x :!^{n_5}(!^{n_1} o \multimap !^{n_1} o), x :!^{n_5}(!^{n_1} o \multimap !^{n_1} o) \cdots \vdash \cdots}{x :!^{n_5}(!^{n_1} o \multimap !^{n_1} o) \cdots \vdash \cdots}\ contr?$$

2. Contraction (remember that in EAL it is allowed only for modal formulas)

$$\Rightarrow \qquad n_5 \geq 1$$

Any n_i ranges over $\mathbb{N}$.

Extension to Linear Logic

- Extension of type inference algorithm to Linear Logic.

Logical Rules	Constraints
$\dfrac{\Gamma, !^{x} A \vdash B}{\Gamma, !^{x-(d+b)} A \vdash B}\ \delta$	$\begin{cases} d + b \leq x - 1 \\ d \geq 0 \\ -1 \leq b \leq 0 \end{cases}$
$\dfrac{\Gamma, A \vdash B}{\Gamma, !^{e} A \vdash B}\ \epsilon$	$e \geq 0$

Extension to Linear Logic

Different variables for different exponential rules

- n_i for !-rules,
- d_i, b_i for δ-rules,
- e_i for ϵ-rules.

Extension to Linear Logic

We can use Linear Programming for minimizing an objective function

$\rightarrow$ find minimal decorations:
$\min \sum_i n_i + \sum_j (b_j + d_j) + \sum_h e_h$

$\rightarrow$ minimal use of δ, ϵ-rules:
$\min \sum_j (b_j + d_j) + \sum_h e_h$

$\Rightarrow$ minimal use of brackets and croissants in Lamping's Optimal Reduction Algorithm

Type Inference

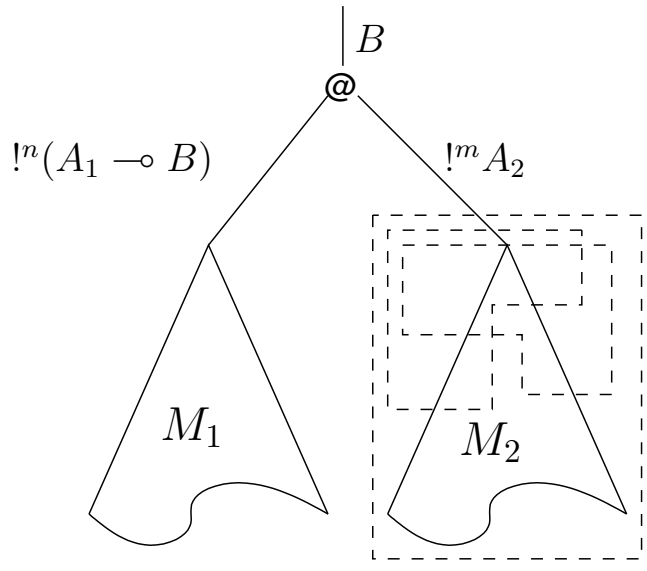
We use NEAL, the natural deduction version of EAL.

$\Rightarrow$ We consider the syntax tree of the term during decoration process (Curry-Howard isomorphism)

$\Rightarrow$!-introduction rule $\leftrightsquigarrow$ to box a piece of graph

Type Inference of application

1. add all possible boxes around to the argument
2. impose type functional (i.e. $n = 0$)
3. unify types for correct application ($A_1 = !^m A_2$)

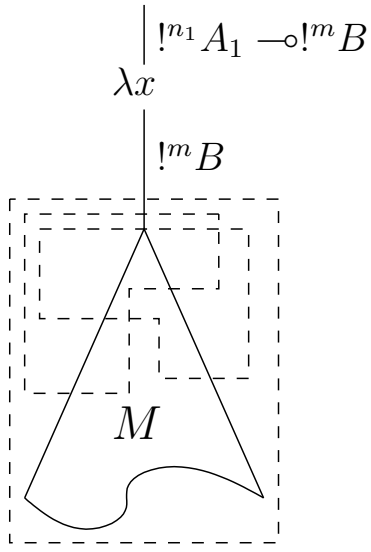


Type Inference of abstraction

1. add all possible boxes around to the body
2. contract types $!^{n_1}A_1, \ldots, !^{n_k}A_k$ of the binded variable

   $\Rightarrow\ n_1 \geq 1$

   $\Rightarrow\ !^{n_1}A_1 = \cdots = !^{n_k}A_k$



Type Inference Algorithm

- Let $M : \sigma$ be a simply typed λ-term
  $\Rightarrow \mathscr{S}(M : \sigma) = \langle \Theta, \Gamma, A \rangle$
  - Θ is a general EAL-type (parametrized number of exponentials),
  - Γ is a base
  - A is a set of linear constraints

X integral solution of $A \Leftrightarrow X(\Gamma) \vdash_{\mathsf{EAL}} M : X(\Theta)$

Principal Typing

- All EAL-types of a λ-term can be obtained from the simple principal type via substitution and solution of the set A of linear constraints, but

Principal Typing

- All EAL-types of a λ-term can be obtained from the simple principal type via substitution and solution of the set A of linear constraints, but
- Type inference algorithm does not say anything on the structure of decorations
- the information is hidden in the set of linear constraints

Principal Typing

$$M ::= x \mid \lambda x.M \mid (M\ M) \mid !\,(M)\left[{}^{M}/_{x},\ldots,{}^{M}/_{x}\right] \mid \|M\|^{M}_{x,y}$$

$$\frac{}{\Gamma, x : A \vdash_{\mathsf{NEAL}} x : A}\ ax \qquad \frac{\Gamma \vdash_{\mathsf{NEAL}} M :!A \quad \Delta, x :!A, y :!A \vdash_{\mathsf{NEAL}} N : B}{\Gamma, \Delta \vdash_{\mathsf{NEAL}} \|N\|^{M}_{x,y} : B}\ contr$$

$$\frac{\Gamma, x : A \vdash_{\mathsf{NEAL}} M : B}{\Gamma \vdash_{\mathsf{NEAL}} \lambda x.M : A \multimap B}\ (\multimap I) \qquad \frac{\Gamma \vdash_{\mathsf{NEAL}} M : A \multimap B \quad \Delta \vdash_{\mathsf{NEAL}} N : A}{\Gamma, \Delta \vdash_{\mathsf{NEAL}} (M\ N) : B}\ (\multimap E)$$

$$\frac{\Delta_1 \vdash_{\mathsf{NEAL}} M_1 :!A_1 \quad \cdots \quad \Delta_n \vdash_{\mathsf{NEAL}} M_n :!A_n \quad x_1 : A_1, \ldots, x_n : A_n \vdash_{\mathsf{NEAL}} N : B}{\Gamma, \Delta_1, \ldots, \Delta_n \vdash_{\mathsf{NEAL}} !\,(N)\left[{}^{M_1}/_{x_1},\ldots,{}^{M_n}/_{x_n}\right] :!B}\ !$$

EA-types for Λ

Definition Let $M \in \Lambda$ and let $EA_M = \{N \mid N \in \Lambda^{EA}$, N is simple and $(N)^* = M\}$.

Then $\Gamma \vdash_{\mathsf{NEAL}} M : A$ if and only if $\Gamma \vdash_{\mathsf{NEAL}} P : A$, for some $P \in EA_M$. EA_M is the set of EA-terms corresponding to M.

EA-types for Λ

Definition Let $M \in \Lambda$ and let $EA_M = \{N \mid N \in \Lambda^{EA}$, N is simple and $(N)^* = M\}$.
Then $\Gamma \vdash_{\mathsf{NEAL}} M : A$ if and only if $\Gamma \vdash_{\mathsf{NEAL}} P : A$, for some $P \in EA_M$. EA_M is the set of EA-terms corresponding to M.

Example For every $n \geq 0$, the EA-term 2_n, defined as:

$$\lambda x.[\lambda y.\underbrace{!(!(...!}_{n+1}(x_3(x_4y_1))[^{x_5}/_{x_3}, ^{x_6}/_{x_4}, ^{y_2}/_{y_1}]...)[^{x_1}/_{x_{2n-1}}, ^{x_2}/_{x_{2n}}, ^{y}/_{y_n}]]_{x=x_1,x_2}$$

belongs to $EA_{\lambda xy.(x(xy))}$.

Abstract EA-terms Abs^{EA}

For any M, EA_M is infinite

$$[M]_{N_1 \to (\widehat{x_1}),\ldots,N_k \to (\widehat{x_k})} \mid \nabla(M)[{}^{N_1}/x_1, \cdots, {}^{N_k}/x_n]$$

- $emb : \Lambda^{EA} \to Abs^{EA}$

$$\Gamma \vdash_{\mathsf{NEAL}} M : A \Rightarrow \Gamma \vdash_{abs} emb(M) : A$$

- $bme : Abs^{EA} \to \Lambda^{EA}$

$$\forall M \in Abs^{EA} \quad \Gamma \vdash_{abs} M : A \Rightarrow \exists M' \in bme(M)\ \Gamma \vdash_{\mathsf{NEAL}} M' : A$$

Reduction relation $\rightarrow_{\mathscr{C}an}$

$$\nabla\big(\nabla(M)[y_1/x_1,\cdots,y_n/x_n]\big)[^{M_1}/y_1,\cdots,^{M_n}/y_n]$$
$$\rightarrow_{\mathscr{C}an} \nabla(M)\left[^{M_1}/x_1,\ldots,^{M_n}/x_n\right]$$

$$\left[[M]_{\ldots,x_i\rightarrow(\widehat{y^{n_i}}),\ldots}\right]_{\ldots,N_j\rightarrow(z_1^j,\ldots,z_{k-1}^j,x_i,z_{k+1}^j,\ldots,z_{m_j}^j),\ldots}$$
$$\rightarrow_{\mathscr{C}an} \left[[M]_{\ldots}\right]_{\ldots,N_j\rightarrow(z_1^j,\ldots,z_{k-1}^j,\widehat{y^{n_i}},z_{k+1}^j,\ldots,z_{m_j}^j),\ldots}$$

Lemma [Subject reduction] Let $M \in Abs^{EA}$ and $M \rightarrow^*_{\mathscr{C}an} N$, then

$$\Gamma \vdash_{abs} M : A \Rightarrow \Gamma \vdash_{abs} N : A.$$

Canonical forms

Let us call canonical forms the EA-terms in normal forms with respect to the reduction $\rightarrow_{\mathscr{C}an}$.

Lemma For every $M \in \Lambda$, let $C(M) = \{N \mid N$ is canonic and simple and $(N)^* = M\}$. Then $\Gamma \vdash_{\mathsf{NEAL}} M : \sigma$ if and only if $\Gamma \vdash_{abs} P : \sigma$, for some $P \in C(M)$.

Lemma For every $M \in \Lambda$, $C(M)$ is finite.

Scheme substitutions

- Type schemata $\sigma ::= \alpha \mid \sigma \multimap \sigma \mid !^{p}(\sigma)$;
  - exponential $p ::= n \mid p + p$
- Scheme substitution

$$\langle S, X \rangle(\alpha) = S(\alpha);$$

$$\langle S, X \rangle(\sigma \multimap \tau) = \langle S, X \rangle(\sigma) \multimap \langle S, X \rangle(\tau);$$

$$\langle S, X \rangle(!^{n_1 + \ldots + n_i}\sigma) = \underbrace{!\ldots!}_{q}\langle S, X \rangle(\sigma),$$

where $q = X(n_1) + \ldots + X(n_i)$ and $X(n_i) \geq 1$.

Principal typing for Abs^{EA}

- Syntax directed
- Let $M \in Abs^{EA}$

$$\Rightarrow \qquad PT(M) = \langle \Theta, \Gamma, A \rangle$$

$$\langle S, X \rangle \text{ such that } X \text{ satisfies } A$$

$$\Updownarrow$$

$$\langle S, X \rangle(\Gamma) \vdash_{abs} M : \langle S, X \rangle(\Theta)$$

Principal typing for Λ in EAL

Let M pure λ-term

$\Rightarrow \mathscr{C}(M)$

$\Rightarrow PT(N) = \langle \Theta_N, \Gamma_N, A_N \rangle$ for any $N \in \mathscr{C}(M)$

$\langle S_N, X_N \rangle$ such that X_N satisfies A_N

$\Updownarrow$

$\langle S_N, X_N \rangle(\Gamma_N) \vdash_{\mathsf{NEAL}} M : \langle S_N, X_N \rangle(\Theta_N)$

Canonical forms give information on the structure of decorations

Phase semantics

- In (Kanovich, Okada, Scedrov) a definition for LLL based on the concept of fibred phase space
- In (Dal Lago, Martini. 2003) a simpler definition based on extensions of the usual one for LL
- Phase semantics for EAL can be obtained modifying the classical notion of phase space
  - introducing a monoidal endomorphism $\phi : M \to M$
  - imposing $\perp \subseteq M^{\perp}$

Phase semantics

- Strong Completeness using the same technique of Okada
- decidability following proof of decidability of LLW by Lafont
- All the results also hold for LAL, IEAL and ILAL