$\textbf{Church} \Rightarrow \textbf{Scott} = \textbf{Ptime}$: an application of resource-sensitive realizability

Kazushige Terui

RIMS, Kyoto University

email: terui@kurims.kyoto-u.ac.jp

Aloïs Brunel

ENS Paris

Background 1

Realizability:

- A way to extract useful information from proofs.
- Define a binary predicate $t \Vdash A$ by induction on formulas:

$$t \Vdash A \Rightarrow B \quad \text{iff} \quad \forall u. \quad u \Vdash A \Longrightarrow tu \Vdash B$$

- Prove adequacy by induction on proofs:

$$\begin{array}{c} \vdots\ \pi \\ \vdash A \end{array} \qquad \Longrightarrow \qquad \pi^* \Vdash A$$

- Related methods: logical relations, Tait-Girard reducibility argument, ...

Background 2

Linear implication $L \multimap A$ vs Intuitionistic implication $A \Rightarrow B$.

In the Curry-Howard setting,

$\lambda x.t : L_1 \multimap L_2$ roughly when x occurs at most once in t.

Notice: In this talk, ‘linear” actually means “affine.”

Background 2

Linear implication $L \multimap A$ vs Intuitionistic implication $A \Rightarrow B$.

In the Curry-Howard setting,

$$\lambda x.t : L_1 \multimap L_2 \quad \text{roughly when } x \text{ occurs at most once in } t.$$

Notice: In this talk, ‘linear” actually means “affine.”

Fundamental question: How do you distinguish $\multimap$ from $\Rightarrow$ in realizability semantics?

Background 2

Linear implication $L \multimap A$ vs Intuitionistic implication $A \Rightarrow B$.

In the Curry-Howard setting,

$$\lambda x.t : L_1 \multimap L_2 \quad \text{roughly when } x \text{ occurs at most once in } t.$$

Notice: In this talk, ‘linear” actually means “affine.”

Fundamental question: How do you distinguish $\multimap$ from $\Rightarrow$ in realizability semantics?

Key: When x is linear in t,

$$\mathbf{Cost}((\lambda x.t)u) \leq \mathbf{Cost}(\lambda x.t) + \mathbf{Cost}(u) + c.$$

Background 3

Resource-sensitive realizability (Hofmann-Dal Lago):

Define a ternary relation

$$t, p \Vdash A$$

- t: realizer = program.
- p: majorizer bounding the cost of t.

The adequacy theorem (or the “basic lemma”) states:

$$\vdash t : A \quad \Longrightarrow \quad t, p \Vdash A \text{ for some } p.$$

Based on this, HD prove Ptime soundness for LAL, LFPL, SAL, BLL.

Background 4

Lambda-calculus characterization of Ptime (Leivant-Marion 93):

Consider $\lambda_{\rightarrow,\times}$ with constants:

$$\epsilon : o, \qquad s_0, s_1, p : o \rightarrow o, \qquad dscr : o \rightarrow o^3 \rightarrow o$$

We have two representations of binary word 010:

First order	$s_0(s_1(s_0(\epsilon)))$	$: o$
Church	$\lambda f_0 f_1 x. f_0(f_1(f_0 x))$	$: \mathsf{W}^\bullet(\alpha)$,

where $\mathsf{W}^\bullet(\alpha) := (\alpha \rightarrow \alpha)^2 \rightarrow (\alpha \rightarrow \alpha)$.

Theorem: $f : \{0,1\}^* \longrightarrow \{0,1\}^*$ is Ptime if and only if it is represented by a term of type $\mathsf{W}^\bullet(o^m) \rightarrow o$ for some m.

Background 5

Church numerals:

$$\mathsf{n}^\bullet \equiv \lambda f x.\ \underbrace{f(\ldots f}_{n\ times}(x)\ldots) \quad \text{have type} \quad \mathsf{N}^\bullet \equiv \forall\alpha(\alpha \Rightarrow \alpha) \Rightarrow (\alpha \Rightarrow \alpha).$$

Background 5

Church numerals:

$$\mathsf{n}^\bullet \equiv \lambda f x.\ \underbrace{f(\ldots f}_{n\ times}(x)\ldots) \quad \text{have type} \quad \mathsf{N}^\bullet \equiv \forall\alpha(\alpha \Rightarrow \alpha) \Rightarrow (\alpha \Rightarrow \alpha).$$

Where do they come from?

$$N \quad = \quad \bigcap\{\alpha : 0 \in \alpha,\ \forall x.(x \in \alpha \Rightarrow x+1 \in \alpha)\}$$

$$n \in N \quad \equiv \quad \forall \alpha.\forall x.(x \in \alpha \Rightarrow x+1 \in \alpha) \Rightarrow 0 \in \alpha \Rightarrow n \in \alpha$$

Background 5

Church numerals:

$$\mathsf{n}^\bullet \equiv \lambda f x.\ \underbrace{f(...f}_{n\ times}(x)...) \quad \text{have type} \quad \mathsf{N}^\bullet \equiv \forall\alpha(\alpha \Rightarrow \alpha) \Rightarrow (\alpha \Rightarrow \alpha).$$

Where do they come from?

$$N \;=\; \bigcap\{\alpha : 0 \in \alpha,\ \forall x.(x \in \alpha \Rightarrow x+1 \in \alpha)\}$$
$$n \in N \;\equiv\; \forall\alpha.\forall x.(x \in \alpha \Rightarrow x+1 \in \alpha) \Rightarrow 0 \in \alpha \Rightarrow n \in \alpha$$

- By extracting a λ-term from the proof of $n \in N$, we obtain $\mathsf{n}^\bullet$.

Background 5

Church numerals:

$$\mathsf{n}^\bullet \equiv \lambda f x.\ \underbrace{f(\ldots f}_{n\ times}(x)\ldots) \quad \text{have type} \quad \mathsf{N}^\bullet \equiv \forall\alpha(\alpha \Rightarrow \alpha) \Rightarrow (\alpha \Rightarrow \alpha).$$

Where do they come from?

$$N = \bigcap\{\alpha : 0 \in \alpha,\ \forall x.(x \in \alpha \Rightarrow x+1 \in \alpha)\}$$

$$n \in N \equiv \forall\alpha.\forall x.(x \in \alpha \Rightarrow x+1 \in \alpha) \Rightarrow 0 \in \alpha \Rightarrow n \in \alpha$$

- By extracting a λ-term from the proof of $n \in N$, we obtain $\mathsf{n}^\bullet$.
- $n \in N$ can be simplified to $\mathsf{N}^\bullet$:

$$n \in N \equiv \forall\alpha.\forall x.(x \in \alpha \Rightarrow x+1 \in \alpha) \Rightarrow 0 \in \alpha \Rightarrow n \in \alpha$$

$$\mathsf{N}^\bullet \equiv \forall\alpha(\alpha \Rightarrow \alpha) \Rightarrow (\alpha \Rightarrow \alpha).$$

Background 5

Scott numerals:

$$\begin{aligned} 0^\circ &\equiv \lambda xy.y \\ \mathsf{n}+1^\circ &\equiv \lambda xy.x(\mathsf{n}^\circ) \end{aligned}$$

have type $\mathsf{N}^\circ = \forall\alpha.(\mathsf{N}^\circ \multimap \alpha) \multimap (\alpha \multimap \alpha)$.

Background 5

Scott numerals:

$$\begin{array}{rcl} \mathsf{0}^\circ & \equiv & \lambda xy.y \\ \mathsf{n}+1^\circ & \equiv & \lambda xy.x(\mathsf{n}^\circ) \end{array}$$

have type $\mathsf{N}^\circ = \forall\alpha.(\mathsf{N}^\circ \multimap \alpha) \multimap (\alpha \multimap \alpha)$.

Where do they come from?

$$n \in N' \quad \equiv \quad n = 0 \vee \exists x \in N'.n = x + 1$$

Background 5

Scott numerals:

$$0^\circ \equiv \lambda xy.y \qquad n+1^\circ \equiv \lambda xy.x(n^\circ)$$

have type $N^\circ = \forall\alpha.(N^\circ \multimap \alpha) \multimap (\alpha \multimap \alpha)$.

Where do they come from?

$$n \in N' \equiv n = 0 \vee \exists x \in N'.n = x + 1$$

Two solutions:

$$N' = \mathbb{N} \quad \text{or} \quad N' = \mathbb{N} \cup \{\omega\}$$

We do not specify which N' is. Still $n \in N'$ is provable.

Background 5

By noting $A \vee B \equiv \forall \alpha.(A \Rightarrow \alpha) \Rightarrow (B \Rightarrow \alpha) \Rightarrow \alpha$,

$$n \in N' \equiv \forall \alpha.(\exists x \in N'.n = x + 1 \Rightarrow \alpha) \Rightarrow (x = 0 \Rightarrow \alpha) \Rightarrow \alpha$$

and one extracts Scott numeral n° from the proof of $n \in N'$.
$n \in N'$ simplifies to N°:

$$\begin{aligned} n \in N' &\equiv \forall \alpha.(\exists x \in N'.n = x + 1 \Rightarrow \alpha) \Rightarrow (x = 0 \Rightarrow \alpha) \Rightarrow \alpha \\ N'' &\equiv \forall \alpha.(N'' \Rightarrow \alpha) \Rightarrow (\alpha \Rightarrow \alpha) \\ \mathsf{N}^\circ &\equiv \forall \alpha.(\mathsf{N}^\circ \multimap \alpha) \multimap (\alpha \multimap \alpha). \end{aligned}$$

Background 5

By noting $A \vee B \equiv \forall\alpha.(A \Rightarrow \alpha) \Rightarrow (B \Rightarrow \alpha) \Rightarrow \alpha$,

$$n \in N' \equiv \forall\alpha.(\exists x \in N'.n = x + 1 \Rightarrow \alpha) \Rightarrow (x = 0 \Rightarrow \alpha) \Rightarrow \alpha$$

and one extracts Scott numeral n° from the proof of $n \in N'$.
$n \in N'$ simplifies to N°:

$$\begin{aligned} n \in N' &\equiv \forall\alpha.(\exists x \in N'.n = x + 1 \Rightarrow \alpha) \Rightarrow (x = 0 \Rightarrow \alpha) \Rightarrow \alpha \\ N'' &\equiv \forall\alpha.(N'' \Rightarrow \alpha) \Rightarrow (\alpha \Rightarrow \alpha) \\ \mathsf{N}^\circ &\equiv \forall\alpha.(\mathsf{N}^\circ \multimap \alpha) \multimap (\alpha \multimap \alpha). \end{aligned}$$

Background 5

By noting $A \vee B \equiv \forall\alpha.(A \Rightarrow \alpha) \Rightarrow (B \Rightarrow \alpha) \Rightarrow \alpha$,

$$n \in N' \equiv \forall\alpha.(\exists x \in N'.n = x + 1 \Rightarrow \alpha) \Rightarrow (x = 0 \Rightarrow \alpha) \Rightarrow \alpha$$

and one extracts Scott numeral n° from the proof of $n \in N'$.
$n \in N'$ simplifies to N°:

$$\begin{aligned} n \in N' &\equiv \forall\alpha.(\exists x \in N'.n = x + 1 \Rightarrow \alpha) \Rightarrow (x = 0 \Rightarrow \alpha) \Rightarrow \alpha \\ N'' &\equiv \forall\alpha.(N'' \Rightarrow \alpha) \Rightarrow (\alpha \Rightarrow \alpha) \\ \mathsf{N}^\circ &\equiv \forall\alpha.(\mathsf{N}^\circ \multimap \alpha) \multimap (\alpha \multimap \alpha). \end{aligned}$$

- n° is a linear λ-term.
- Does not support recursion by itself, but admits a natural definition of predecessor and discriminator.

Outline

- We replace the first order words of Leivant-Marion by Scott words.

Outline

- We replace the first order words of Leivant-Marion by Scott words.
- For this, we introduce a variant of linear logic ($\Rightarrow$, $\multimap$) with second order quantifier $\forall$ and type fixpoint operator μ, both restricted to linear formulas.

Outline

- We replace the first order words of Leivant-Marion by Scott words.
- For this, we introduce a variant of linear logic ($\Rightarrow, \multimap$) with second order quantifier $\forall$ and type fixpoint operator μ, both restricted to linear formulas.
- We prove: a function $f : \{0,1\}^* \longrightarrow \{0,1\}^*$ is Ptime if and only if it is represented by a term of type $Church \Rightarrow Scott$.

Outline

- We replace the first order words of Leivant-Marion by Scott words.
- For this, we introduce a variant of linear logic ($\Rightarrow, \multimap$) with second order quantifier $\forall$ and type fixpoint operator μ, both restricted to linear formulas.
- We prove: a function $f : \{0,1\}^* \longrightarrow \{0,1\}^*$ is Ptime if and only if it is represented by a term of type $Church \Rightarrow Scott$.
- To prove Ptime soundness we employ resource sensitive realizability (after Hofmann-Dal Lago).

$$t, p \Vdash A$$

- t: Realizer
- p: Majorizer
- A: Formula
- $\Vdash$: Realizability relation

CBV lambda calculus with cost model

Consider the untyped CBV lambda calculus:

$$(\lambda x.t)v \to t[v/x]$$

where v is a value, i.e. an abstraction.

CBV lambda calculus with cost model

Consider the untyped CBV lambda calculus:

$$(\lambda x.t)v \to t[v/x]$$

where v is a value, i.e. an abstraction.

Difficulty: Cost of one-step is not constant.

CBV lambda calculus with cost model

Consider the untyped CBV lambda calculus:

$$(\lambda x.t)v \to t[v/x]$$

where v is a value, i.e. an abstraction.

Difficulty: Cost of one-step is not constant.

Hence we explicitly mention the cost of reduction (Dal-Lago, Martini 2008):

$$t \xrightarrow{n} u, \quad \text{if } t \to u \text{ and } n = max\{|u| - |t|, 1\}$$

CBV lambda calculus with cost model

Consider the untyped CBV lambda calculus:

$$(\lambda x.t)v \to t[v/x]$$

where v is a value, i.e. an abstraction.

Difficulty: Cost of one-step is not constant.

Hence we explicitly mention the cost of reduction (Dal-Lago, Martini 2008):

$$t \stackrel{n}{\to} u, \qquad \text{if } t \to u \text{ and } n = max\{|u| - |t|, 1\}$$

Fact: Suppose that $(\lambda x.t)v \stackrel{n}{\to} t[v/x]$ and x occurs c times in t. Then

$$\begin{aligned} n &= 1 && \text{if } c \leq 1 \\ n &\leq (c-1)|v| && \text{if } c > 1. \end{aligned}$$

CBV lambda calculus with cost model

Definition: When $t \to^* v$,

- $[\![t]\!] = v$
- $\mathbf{Cost}(t) := |t| + n$ where $t \overset{n}{\twoheadrightarrow} v$.

Theorem (Dal Lago-Martini 2008): There exists a Turing machine M_{eval} that p-simulates CBV lambda calculus: given a (converging) λ-term t with $\mathbf{Cost}(t) = n$, M_{eval} computes $[\![t]\!]$ in time $O(n^4)$.

$$t, p \Vdash A$$

- t: Realizer
- p: Majorizer
- A: Formula
- $\Vdash$: Realizability relation

The dual type system $\mathbf{DIAL}_{lin}$

$\mathbf{DIAL}_{lin}$ = Dual Intuitionistic Affine Logic consists of formulas

$$L \multimap A, \quad A \Rightarrow B, \quad \forall\alpha.A, \quad \mu\alpha.L.$$

- The $\multimap$-fragment is affine logic (i.e. FL_{ew})
- The $\Rightarrow$-fragment is intuitionistic logic
- $\Rightarrow$ dominates $\multimap$:

$$\frac{L \multimap A}{L \Rightarrow A}$$

- $\forall, \mu$ are restricted to affine formulas (i.e. those without $\Rightarrow$):

$$\forall\alpha.A(\alpha) \multimap A(L), \qquad \mu\alpha.L(\alpha) \circ\!\!-\!\!\circ L(\mu\alpha.L(\alpha))$$

(Note: $\mu\alpha.L$ can be any fixed point.)

The dual type system $\mathbf{DIAL}_{lin}$

Linear and general formulas:

$$\begin{aligned} L &::= \alpha \mid \forall\alpha L \mid \mu\alpha L^{(*)} \mid L \multimap L, \\ A &::= L \mid \forall\alpha A \mid L \multimap A \mid A \Rightarrow A. \end{aligned}$$

$(*)$: α occurs only positively in L.

Judgment: $\Gamma \; ; \; \Delta \vdash t : A$, where

- Δ consists of $x : L$ with L a linear formula, and
- Γ consists of $x : A$ with A an arbitrary formula.

The dual type system $\mathbf{DIAL}_{lin}$

$$\frac{}{x : A\ ;\ \vdash x : A}\ (ax1) \qquad \frac{}{;\ x : L \vdash x : L}\ (ax2)$$

$$\frac{\Gamma\ ;\ \Delta \vdash t : \mu\alpha L}{\Gamma\ ;\ \Delta \vdash t : L[\mu\alpha L/\alpha]}\ (\mu_e) \qquad \frac{\Gamma\ ;\ \Delta \vdash t : L[\mu\alpha L/\alpha]}{\Gamma\ ;\ \Delta \vdash t : \mu\alpha L}\ (\mu_i)$$

$$\frac{\Gamma\ ;\ \Delta \vdash t : A \qquad \alpha \notin FV(\Gamma\ ;\ \Delta)}{\Gamma\ ;\ \Delta \vdash t : \forall\alpha A}\ (\forall_i) \qquad \frac{\Gamma\ ;\ \Delta \vdash t : \forall\alpha A}{\Gamma\ ;\ \Delta \vdash t : A[L/\alpha]}\ (\forall_e)$$

The dual type system $\mathbf{DIAL}_{lin}$

$$\frac{\Gamma_1\ ;\ \Delta \vdash t : A \Rightarrow B \qquad \Gamma_2\ ;\ \vdash u : A}{\Gamma_1, \Gamma_2\ ;\ \Delta \vdash tu : B}\ (\Rightarrow_e) \qquad \frac{\Gamma, z : A\ ;\ \Delta \vdash t : B}{\Gamma\ ;\ \Delta \vdash \lambda z.t : A \Rightarrow B}\ (\Rightarrow_i)$$

$$\frac{\Gamma_1\ ;\ \Delta_1 \vdash t : L \multimap B \qquad \Gamma_2\ ;\ \Delta_2 \vdash u : L}{\Gamma_1, \Gamma_2\ ;\ \Delta_1, \Delta_2 \vdash tu : B}\ (\multimap_e) \qquad \frac{\Gamma\ ;\ \Delta, z : L \vdash t : B}{\Gamma\ ;\ \Delta \vdash \lambda z.t : L \multimap B}\ (\multimap_i)$$

$$\frac{\Gamma, x : A, y : A\ ;\ \Delta \vdash t : B}{\Gamma, z : A\ ;\ \Delta \vdash t[z/x, z/y] : B}\ (Contr) \qquad \frac{\Gamma\ ;\ \Delta, x : L \vdash t : B}{\Gamma, x : L\ ;\ \Delta \vdash t : B}\ (Derel)$$

$$\frac{\Gamma\ ;\ \Delta \vdash t : B}{\Gamma, \Gamma'\ ;\ \Delta, \Delta' \vdash t : B}\ (Weak)$$

Church and Scott data types

Church numerals and words :

$$\mathsf{N}^\bullet \equiv \forall\alpha(\alpha \multimap \alpha) \Rightarrow (\alpha \multimap \alpha)$$

$$\mathsf{n}^\bullet \equiv \lambda f x.\ \underbrace{f(\ldots f}_{n\ times}(x)\ldots)$$

$$\mathsf{mult}^\bullet \equiv \lambda xy\lambda f.x(yf) : \mathsf{N}^\bullet \Rightarrow \mathsf{N}^\bullet \Rightarrow \mathsf{N}^\bullet$$

$$\mathsf{mon}_n^\bullet \equiv \lambda x\lambda f.\ \underbrace{x(\cdots(x}_{n\ times}\ f)\cdots) : \mathsf{N}^\bullet \Rightarrow \mathsf{N}^\bullet$$

$$\mathsf{W}^\bullet \equiv \forall\alpha(\alpha \multimap \alpha) \Rightarrow (\alpha \multimap \alpha) \Rightarrow (\alpha \multimap \alpha)$$

$$\mathsf{w}^\bullet \equiv \lambda f_0.\lambda f_1.\lambda x.f_{i_1}(f_{i_2}(\ldots(f_{i_n}(x)\ldots)))$$

$$(w = i_1 \cdots i_n)$$

Church and Scott data types

Scott numerals and words :

$$\mathsf{N}^\circ \equiv \mu\beta\forall\alpha(\beta \multimap \alpha) \multimap (\alpha \multimap \alpha)$$

$$0^\circ = \lambda xy.y$$

$$(\mathsf{n}+1)^\circ \equiv \lambda xy.x(\mathsf{n}^\circ)$$

$$\mathsf{succ}^\circ \equiv \lambda z.\lambda xy.xz : \mathsf{N}^\circ \multimap \mathsf{N}^\circ$$

$$\mathsf{pred}^\circ \equiv \lambda z.z(\lambda x.x)(0^\circ) : \mathsf{N}^\circ \multimap \mathsf{N}^\circ$$

$$\mathsf{W}^\circ \equiv \mu\beta\forall\alpha(\beta \multimap \alpha) \multimap (\beta \multimap \alpha) \multimap (\alpha \multimap \alpha)$$

$$\epsilon^\circ \equiv \lambda xyz.z$$

$$(0\mathsf{w})^\circ \equiv \lambda xyz.x(\mathsf{w}^\circ)$$

$$(1\mathsf{w})^\circ \equiv \lambda xyz.y(\mathsf{w}^\circ)$$

Church and Scott data types

Finite sets and tensor product :

$$\mathsf{B}_n^\circ \equiv \forall\alpha.\underbrace{\alpha \multimap \ldots \alpha \multimap}_{n\ times} \alpha \qquad L \otimes M \equiv \forall\alpha.(L \multimap M \multimap \alpha) \multimap \alpha$$

$$\mathsf{b}_i^\circ \equiv \lambda x_0 \cdots x_{n-1}.x_i \qquad t \otimes u \equiv \lambda x.xtu \quad (t : L,\ u : M)$$

Decomposer and iteration :

$$\mathsf{dec}^\circ = \lambda z.z(\lambda y.\mathsf{b}_0^\circ \otimes y)(\lambda y.\mathsf{b}_1^\circ \otimes y)(\mathsf{b}_2^\circ \otimes \epsilon^\circ) \ : \ \mathsf{W}^\circ \multimap \mathsf{B}_3^\circ \otimes \mathsf{W}^\circ$$

$$\mathsf{iter}^\bullet = \lambda xfg.xfg \ : \ \mathsf{N}^\bullet \Rightarrow (L \multimap L) \Rightarrow (L \multimap L)$$

FP-completeness

Theorem: Let $f : \{0,1\}^* \to \{0,1\}^*$. T.f.a.e.

1. f is a Ptime function
2. There is a λ-term f : $\mathsf{W}^\bullet \Rightarrow \mathsf{W}^\circ$ in $\mathbf{DIAL}_{lin}$ such that

$$f(w_1) = w_2 \iff \mathsf{f}\mathsf{w}_1^\bullet \longrightarrow_\beta^* \mathsf{w}_2^\circ.$$

(1$\Rightarrow$ 2) is routine.

We prove (2$\Rightarrow$ 1) by realizability.

$$t, p \Vdash A$$

- t: Realizer
- p: Majorizer
- A: Formula
- $\Vdash$: Realizability relation

Majorizers

Consider simple types over base type o.

A higher order additive term p is a λ-term built from constants

$$n \quad : \quad o \qquad \text{(for every natural number } n\text{)}$$

$$+ \quad : \quad o \to o \to o.$$

Identified under $\alpha\beta\eta$- and arithmetical equivalences.

Majorizers

Consider simple types over base type o.

A higher order additive term p is a λ-term built from constants

$$\begin{array}{lll} n & : & o \qquad \text{(for every natural number } n\text{)} \\ + & : & o \to o \to o. \end{array}$$

Identified under $\alpha\beta\eta$- and arithmetical equivalences.

Mapping of $\mathbf{DIAL}_{lin}$ formulas to simple types:

$$o(L) = o, \qquad o(A \Rightarrow B) = o(A) \to o(B), \qquad o(\forall\alpha\, A) = o(A).$$

$t : A$ will be mojorized by $p : o(A)$.

$$\begin{array}{lllll} o(\mathsf{N}^\bullet) & = & o(\forall\alpha(\alpha \multimap \alpha) \Rightarrow (\alpha \multimap \alpha)) & = & o \to o \\ o(\mathsf{W}^\bullet) & = & o(\forall\alpha(\alpha \multimap \alpha) \Rightarrow (\alpha \multimap \alpha) \Rightarrow (\alpha \multimap \alpha)) & = & o \to o \to o \end{array}$$

Saturated sets

A nonempty set $X \subseteq \Lambda \times \mathbb{N}$ is a saturated set (of type o) if

(cost) $(t, n) \in X \implies \mathbf{Cost}(t) \leq n$;

(monotonicity) $(t, n) \in X \implies (t, m) \in X$ for every $m \geq n$;

(exchange) $((\lambda xy.t(x, y))vw, n) \in X \implies ((\lambda yx.t(y, x))wv, n) \in X$;

(contraction) $((\lambda xy.t(x, y))vv, n) \in X \implies ((\lambda z.t(z, z))v, n) \in X$;

(identity) $(v, n) \in X \implies ((\lambda x.x)v, n + 3) \in X$;

$\dots$

$$t, p \Vdash A$$

- t: Realizer
- p: Majorizer
- A: Formula
- $\Vdash$: Realizability relation

Realizability relation

A valuation η maps each propositional variable α to a saturated set $\eta(\alpha)$.

$t, p \Vdash_\eta A$, where $p : o(A)$, is defined by induction on A:

- $t,\ n \Vdash_\eta \alpha$ iff $(t,\ n) \in \eta(\alpha)$.
- $t,\ p \Vdash_\eta L \multimap A$ iff $u, m \Vdash_\eta L \Longrightarrow tu, p + m \Vdash_\eta A$ for every u, m, and $\mathbf{Cost}(t) \leq\downarrow p$.
- $t,\ p \Vdash_\eta B \Rightarrow A$ iff $u, q \Vdash_\eta B \Longrightarrow tu, p(q) \Vdash_\eta A$ for every u, q, and $\mathbf{Cost}(t) \leq\downarrow p$.

Adequacy

- $t,\, p \Vdash_\eta \forall\alpha A$ iff $t,\, p \Vdash_{\eta\{\alpha\leftarrow X\}} A$ for every saturated set X.
- $t,\, n \Vdash_\eta \mu\alpha L$ iff $(t, n) \in X$ for every saturated set X such that $\hat{L}_{\eta\{\alpha\leftarrow X\}} \subseteq X$, where $\hat{L}_\eta = \{(t, n) : t, n \Vdash_\eta L\}$.

Adequacy

- $t,\ p \Vdash_\eta \forall\alpha A$ iff $t,\ p \Vdash_{\eta\{\alpha\leftarrow X\}} A$ for every saturated set X.
- $t,\ n \Vdash_\eta \mu\alpha L$ iff $(t, n) \in X$ for every saturated set X such that $\hat{L}_{\eta\{\alpha\leftarrow X\}} \subseteq X$, where $\hat{L}_\eta = \{(t, n) : t, n \Vdash_\eta L\}$.

Adequacy Theorem: If $\vdash t : A$, then $t, p \Vdash A$ for some $p : o(A)$.

Proof: By induction on the length of the proof.

Examples

1. $\lambda fx.fx, 6 \Vdash (L \multimap M) \multimap (L \multimap M)$

Suppose $v, n \Vdash L \multimap M$ and $w, m \Vdash L$.

$(\lambda f.f)v, n+3 \Vdash L \multimap M$

$(\lambda x.x)w, m+3 \Vdash L$

$(\lambda f.f)v((\lambda x.x)w), n+m+6 \Vdash M$

$(\lambda fx.fx)vw, n+m+6 \Vdash M$. Hence

$\lambda fx.fx, 6 \Vdash (L \multimap M) \multimap (L \multimap M)$.

Examples

1. $\lambda fx.fx, 6 \Vdash (L \multimap M) \multimap (L \multimap M)$

Suppose $v, n \Vdash L \multimap M$ and $w, m \Vdash L$.
$(\lambda f.f)v, n + 3 \Vdash L \multimap M$
$(\lambda x.x)w, m + 3 \Vdash L$
$(\lambda f.f)v((\lambda x.x)w), n + m + 6 \Vdash M$
$(\lambda fx.fx)vw, n + m + 6 \Vdash M$. Hence
$\lambda fx.fx, 6 \Vdash (L \multimap M) \multimap (L \multimap M)$.

2. $(L \multimap L \multimap M) \multimap (L \multimap M)$ cannot be realized.

Examples

1. $\lambda fx.fx, 6 \Vdash (L \multimap M) \multimap (L \multimap M)$

Suppose $v, n \Vdash L \multimap M$ and $w, m \Vdash L$.

$(\lambda f.f)v, n + 3 \Vdash L \multimap M$

$(\lambda x.x)w, m + 3 \Vdash L$

$(\lambda f.f)v((\lambda x.x)w), n + m + 6 \Vdash M$

$(\lambda fx.fx)vw, n + m + 6 \Vdash M$. Hence

$\lambda fx.fx, 6 \Vdash (L \multimap M) \multimap (L \multimap M)$.

2. $(L \multimap L \multimap M) \multimap (L \multimap M)$ cannot be realized.

3. $\lambda fx.fxx, \lambda fx.fxx + 9 \Vdash (A \Rightarrow A \Rightarrow B) \Rightarrow A \Rightarrow B$

Ptime soundness

Lemma: For every $w \in \{0,1\}^n$, we have $\mathsf{w}^\bullet, q_n \Vdash \mathsf{W}^\bullet$ with

$$q_n = \lambda z_0 z_1 . n(z_0 + z_1 + 3) + 3 : o^2 \to o.$$

Ptime soundness

Lemma: For every $w \in \{0,1\}^n$, we have $\mathsf{w}^\bullet, q_n \Vdash \mathsf{W}^\bullet$ with

$$q_n = \lambda z_0 z_1 . n(z_0 + z_1 + 3) + 3 : o^2 \to o.$$

Lemma: If $\lambda x.p(x) : (o^2 \to o) \to o$, then $p(q_n)$ is a polynomial in n.

Ptime soundness

Lemma: For every $w \in \{0,1\}^n$, we have $\mathsf{w}^\bullet, q_n \Vdash \mathsf{W}^\bullet$ with

$$q_n = \lambda z_0 z_1 . n(z_0 + z_1 + 3) + 3 : o^2 \to o.$$

Lemma: If $\lambda x.p(x) : (o^2 \to o) \to o$, then $p(q_n)$ is a polynomial in n.

Example: When $p(x) = (x(x00))(x00)$,

$$\begin{aligned} p(q_n) &= (q_n(q_n 00))(q_n 00) \\ &= (q_n(3n+3))(3n+3) \\ &= n(3n+3+3n+3+3)+3 \\ &= O(n^2) \end{aligned}$$

Ptime soundness

Theorem: Let L be a linear formula. If $\vdash \mathsf{f} : \mathsf{W}^\bullet \Rightarrow L$, then there exists a polynomial P such that for every $w \in \{0,1\}^n$, $\mathbf{Cost}(\mathsf{f}\mathsf{w}^\bullet) \le P(n)$.

Ptime soundness

Theorem: Let L be a linear formula. If $\vdash \mathsf{f} : \mathsf{W}^\bullet \Rightarrow L$, then there exists a polynomial P such that for every $w \in \{0,1\}^n$, $\mathbf{Cost}(\mathsf{fw}^\bullet) \leq P(n)$.

Proof: By adequacy,
$\mathsf{f}, \lambda x.p(x) \Vdash \mathsf{W}^\bullet \Rightarrow L$ for some $\lambda x.p(x) : (o^2 \to o) \to o$.
$\mathsf{w}^\bullet, q_n \Vdash \mathsf{W}^\bullet$ by above. Hence
$\mathsf{fw}^\bullet, p(q_n) \Vdash L$, so $\mathbf{Cost}(\mathsf{fw}^\bullet) \leq p(q_n) = P(n)$.

Ptime soundness

Theorem: Let L be a linear formula. If $\vdash$ f : $\mathsf{W}^\bullet \Rightarrow L$, then there exists a polynomial P such that for every $w \in \{0,1\}^n$, $\mathbf{Cost}(\mathsf{fw}^\bullet) \leq P(n)$.

Proof: By adequacy,
f, $\lambda x.p(x) \Vdash \mathsf{W}^\bullet \Rightarrow L$ for some $\lambda x.p(x) : (o^2 \to o) \to o$.
$\mathsf{w}^\bullet, q_n \Vdash \mathsf{W}^\bullet$ by above. Hence
$\mathsf{fw}^\bullet, p(q_n) \Vdash L$, so $\mathbf{Cost}(\mathsf{fw}^\bullet) \leq p(q_n) = P(n)$.

Corollary: Let f : $\mathsf{W}^\bullet \Rightarrow \mathsf{W}^\circ$. For every $w \in \{0,1\}^*$, the β-normal form of $\mathsf{fw}^\bullet$ can be computed in time polynomial in $|w|$.

Final Remark

In resource sensitive realizability, $L \multimap B$ and $A \Rightarrow B$ are distinguished by means of majorizers.

- $L \multimap B$ majorized by first order resources

Final Remark

In resource sensitive realizability, $L \multimap B$ and $A \Rightarrow B$ are distinguished by means of majorizers.

- $L \multimap B$ majorized by first order resources
- $A \Rightarrow B$ majorized by higher order resources

Final Remark

In resource sensitive realizability, $L \multimap B$ and $A \Rightarrow B$ are distinguished by means of majorizers.

- $L \multimap B$ majorized by first order resources
- $A \Rightarrow B$ majorized by higher order resources
- Scott numerals are linear; $\mathsf{n}^\circ, O(n) \Vdash \mathsf{N}^\circ$.

Final Remark

In resource sensitive realizability, $L \multimap B$ and $A \Rightarrow B$ are distinguished by means of majorizers.

- $L \multimap B$ majorized by first order resources
- $A \Rightarrow B$ majorized by higher order resources
- Scott numerals are linear; $\mathsf{n}^\circ, O(n) \Vdash \mathsf{N}^\circ$.
- Church numerals are nonlinear; $\mathsf{n}^\bullet, \lambda x.n(x+3)+3 \Vdash \mathsf{N}^\bullet$. It has a multiplying effect.