Normalization by Evaluation for Call-by-Push-Value

Towards a Modal-Logical Reconstruction of NbE

Andreas Abel[1] Christian Sattler[2]

[1]Department of Computer Science and Engineering
Chalmers and Gothenburg University, Sweden

[2]School of Computer Science
University of Nottingham, UK

25th Int. Conf. on Types for Proofs and Programs
TYPES 2019, Oslo, Norway
13 June 2019

Trailer

- Setting

normalization-by-evaluation	$=$	completeness $\circ$ soundness
soundness	$=$	interpretiation (into any model)
completeness	$=$	reification (from syntactic model)

- Story

Reason	semantics needs	std. solution	our solution
Reification $\Rightarrow$	monotonicity	presheaves	comonad coalgebras
Reflection $+$	case distinction	sheaves	monad algebra

- Suspension: what's call-by-push-value to do with this?

Simply-typed lambda-calculus

Z

$$\frac{x : A \in \Gamma}{\Gamma \vdash x : A}$$

$$\frac{\Gamma \vdash t : A \Rightarrow B \qquad \Gamma \vdash u : A}{\Gamma \vdash t\, u : B}$$

$$\frac{\Gamma, x{:}A \vdash t : B}{\Gamma \vdash \lambda x.\, t : A \Rightarrow B}$$

Simply-typed lambda-calculus

$$\frac{x : A \in \Gamma}{x : \Gamma \vdash A}$$

$$\frac{t : \Gamma \vdash A \Rightarrow B \qquad u : \Gamma \vdash A}{t\, u : \Gamma \vdash B}$$

$$\frac{t : \Gamma, x{:}A \vdash B}{\lambda x.\, t : \Gamma \vdash A \Rightarrow B}$$

Simply-typed lambda-calculus

$$\frac{x : (A \in \Gamma)}{x : (\Gamma \vdash A)}$$

$$\frac{t : (\Gamma \vdash A \Rightarrow B) \qquad u : (\Gamma \vdash A)}{t\,u : (\Gamma \vdash B)}$$

$$\frac{t : (\Gamma, x{:}A \vdash B)}{\lambda x.\, t : (\Gamma \vdash A \Rightarrow B)}$$

Simply-typed lambda-calculus

W

$$\frac{x : (A \in \Gamma)}{\mathsf{var}\ x : (\Gamma \vdash A)}$$

$$\frac{t : (\Gamma \vdash A \Rightarrow B) \qquad u : (\Gamma \vdash A)}{\mathsf{app}\ t\ u : (\Gamma \vdash B)}$$

$$\frac{t : (\Gamma.A \vdash B)}{\mathsf{abs}\ t : (\Gamma \vdash A \Rightarrow B)}$$

Simply-typed lambda-calculus V

$$\frac{x : A \in \Gamma}{\mathsf{var}\ x : \Gamma \vdash A}$$

$$\frac{t : \Gamma \vdash A \Rightarrow B \qquad u : \Gamma \vdash A}{\mathsf{app}\ t\ u : \Gamma \vdash B}$$

$$\frac{t : \Gamma.A \vdash B}{\mathsf{abs}\ t : \Gamma \vdash A \Rightarrow B}$$

Simply-typed lambda-calculus

U

var $\dfrac{A \in \Gamma}{\Gamma \vdash A}$

app $\dfrac{\Gamma \vdash A \Rightarrow B \qquad \Gamma \vdash A}{\Gamma \vdash B}$

abs $\dfrac{\Gamma.A \vdash B}{\Gamma \vdash A \Rightarrow B}$

Simply-typed lambda-calculus

T

var $\dfrac{\mathsf{Var}_{\Gamma}^{A}}{\mathsf{Tm}_{\Gamma}^{A}}$

app $\dfrac{\mathsf{Tm}_{\Gamma}^{A \Rightarrow B} \qquad \mathsf{Tm}_{\Gamma}^{A}}{\mathsf{Tm}_{\Gamma}^{B}}$

abs $\dfrac{\mathsf{Tm}_{\Gamma.A}^{B}}{\mathsf{Tm}_{\Gamma}^{A \Rightarrow B}}$

Simply-typed lambda-calculus

S

$$
\begin{array}{lcll}
\mathsf{var} & : & \mathsf{Var}_\Gamma^A & \to \mathsf{Tm}_\Gamma^A \\
\mathsf{app} & : & \mathsf{Tm}_\Gamma^{A \Rightarrow B} \times \mathsf{Tm}_\Gamma^A & \to \mathsf{Tm}_\Gamma^B \\
\mathsf{abs} & : & \mathsf{Tm}_{\Gamma.A}^B & \to \mathsf{Tm}_\Gamma^{A \Rightarrow B}
\end{array}
$$

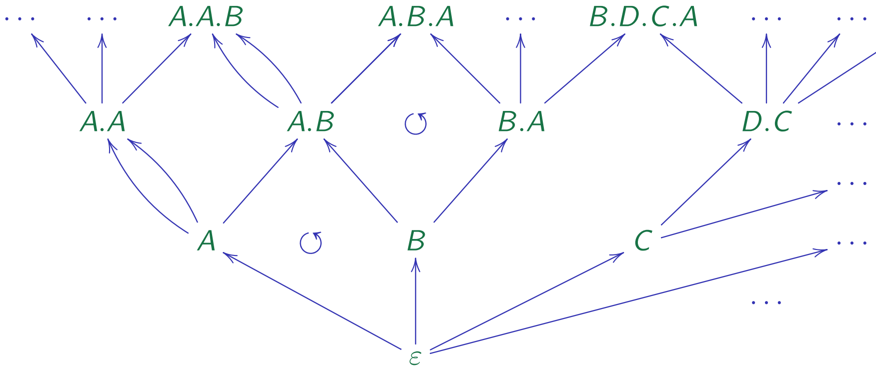
Simply-typed lambda-calculus

R

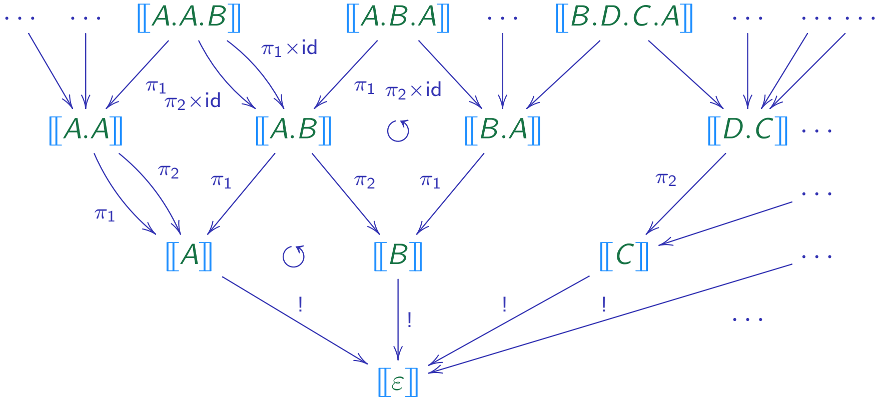
$$
\begin{array}{lcll}
\mathsf{var} & : & \mathsf{Var}^A_\Gamma & \to \mathsf{Tm}^A_\Gamma \\
\mathsf{app} & : & \mathsf{Tm}^{A \Rightarrow B}_\Gamma \times \mathsf{Tm}^A_\Gamma & \to \mathsf{Tm}^B_\Gamma \\
\mathsf{abs} & : & \mathsf{Tm}^B_{\Gamma.A} & \to \mathsf{Tm}^{A \Rightarrow B}_\Gamma
\end{array}
$$

$$
\begin{array}{lcll}
\mathsf{zero} & : & 1 & \to \mathsf{Var}^A_{\Gamma.A} \\
\mathsf{suc} & : & \mathsf{Var}^A_\Gamma & \to \mathsf{Var}^A_{\Gamma.A'}
\end{array}
$$

Contexts $\Gamma \subseteq \Delta$: branching time



Context interpretation $[\![\Gamma]\!]$



$$[\![A_n.\ldots.A_1]\!] = [\![A_n]\!] \dot{\times} \cdots \dot{\times} [\![A_1]\!]$$

Context-indexed sets: temporal propositions!?

$$\mathcal{A} \dot{\to} \mathcal{B} = \forall \Gamma.\ \mathcal{A}_\Gamma \to \mathcal{B}_\Gamma$$ morphism

$$(\mathcal{A} \dot{\times} \mathcal{B})_\Gamma = \mathcal{A}_\Gamma \times \mathcal{B}_\Gamma$$ pointwise constructions

$$(\mathcal{A} \dot{\Rightarrow} \mathcal{B})_\Gamma = \mathcal{A}_\Gamma \to \mathcal{B}_\Gamma$$

$$\vdots$$

$$([A]\,\mathcal{B})_\Gamma = \mathcal{B}_{\Gamma.A}$$ next-time (dynamic logic)

$$(\Box\,\mathcal{B})_\Gamma = \forall \Delta \supseteq \Gamma.\ \mathcal{B}_\Delta$$ forever (AG)

$$(\Diamond \mathcal{B})_\Gamma = \exists \Delta \supseteq \Gamma.\ \mathcal{B}_\Delta$$ sometimes (EF)

$$\Box\,(\mathcal{A} \dot{\Rightarrow} \mathcal{B}) = \forall \Delta \supseteq \Gamma.\ \mathcal{A}_\Delta \to \mathcal{B}_\Delta$$ Kripke function space

Laws for context extension

$$(\mathcal{A} \mathrel{\dot{\to}} \mathcal{B}) \quad \to \quad ([A]\,\mathcal{A} \mathrel{\dot{\to}} [A]\,\mathcal{B}) \qquad \text{functor}$$

$$[A]\,(\mathcal{A} \mathbin{\dot{\times}} \mathcal{B}) \quad = \quad [A]\,\mathcal{A} \mathbin{\dot{\times}} [A]\,\mathcal{B} \qquad \text{distributes}$$

$$[A]\,(\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B}) \quad = \quad [A]\,\mathcal{A} \mathrel{\dot{\Rightarrow}} [A]\,\mathcal{B}$$

$$\vdots$$

Laws forever

$$
\begin{array}{rcll}
(\mathcal{A} \mathrel{\dot\to} \mathcal{B}) & \to & \Box\mathcal{A} \mathrel{\dot\to} \Box\mathcal{B} & \text{functor} \\
\\
\Box\mathcal{B} & \dot\to & \mathcal{B} & \text{comonad} \\
\Box\mathcal{B} & \dot\to & \Box\Box\mathcal{B} & \\
\\
\dot{1} & \dot\to & \Box\dot{1} & \text{monoidality} \\
\Box\mathcal{A} \mathbin{\dot\times} \Box\mathcal{B} & \dot\to & \Box(\mathcal{A} \mathbin{\dot\times} \mathcal{B}) & \\
\\
\Box\mathcal{B} & \dot\to & [A]\,\mathcal{B} & \text{instantiation}
\end{array}
$$

Simply-typed lambda-calculus

R

$$
\begin{array}{lcll}
\mathsf{var} & : & \mathsf{Var}^{A}_{\Gamma} & \to \mathsf{Tm}^{A}_{\Gamma} \\
\mathsf{app} & : & \mathsf{Tm}^{A \Rightarrow B}_{\Gamma} \times \mathsf{Tm}^{A}_{\Gamma} & \to \mathsf{Tm}^{B}_{\Gamma} \\
\mathsf{abs} & : & \mathsf{Tm}^{B}_{\Gamma.A} & \to \mathsf{Tm}^{A \Rightarrow B}_{\Gamma}
\end{array}
$$

$$
\begin{array}{lcll}
\mathsf{zero} & : & 1 & \to \mathsf{Var}^{A}_{\Gamma.A} \\
\mathsf{suc} & : & \mathsf{Var}^{A}_{\Gamma} & \to \mathsf{Var}^{A}_{\Gamma.A'}
\end{array}
$$

Simply-typed lambda-calculus

Q

$$\mathsf{var} : \mathsf{Var}^A_\Gamma \to \mathsf{Tm}^A_\Gamma$$

$$\mathsf{app} : \mathsf{Tm}^{A \Rightarrow B}_\Gamma \times \mathsf{Tm}^A_\Gamma \to \mathsf{Tm}^B_\Gamma$$

$$\mathsf{abs} : \big([A]\,\mathsf{Tm}^B\big)_\Gamma \to \mathsf{Tm}^{A \Rightarrow B}_\Gamma$$

$$\mathsf{zero} : 1 \to \big([A]\,\mathsf{Var}^A\big)_\Gamma$$

$$\mathsf{suc} : \mathsf{Var}^A_\Gamma \to \big([A']\,\mathsf{Var}^A\big)_\Gamma$$

Simply-typed lambda-calculus

P

$$\begin{array}{lcl} \mathsf{var} & : & \mathsf{Var}^A \quad \dot{\to} \quad \mathsf{Tm}^A \\ \mathsf{app} & : & \mathsf{Tm}^{A \Rightarrow B} \mathbin{\dot{\times}} \mathsf{Tm}^A \quad \dot{\to} \quad \mathsf{Tm}^B \\ \mathsf{abs} & : & [A]\, \mathsf{Tm}^B \quad \dot{\to} \quad \mathsf{Tm}^{A \Rightarrow B} \end{array}$$

$$\begin{array}{lcl} \mathsf{zero} & : & 1 \quad \dot{\to} \quad [A]\, \mathsf{Var}^A \\ \mathsf{suc} & : & \mathsf{Var}^A \quad \dot{\to} \quad [A']\, \mathsf{Var}^A \end{array}$$

Normal forms

N

$$\mathsf{var} \;:\; \mathsf{Var}^{A} \;\dot{\to}\; \mathsf{Ne}^{A}$$

$$\mathsf{app} \;:\; \mathsf{Ne}^{A \Rightarrow B} \mathbin{\dot{\times}} \mathsf{Nf}^{A} \;\dot{\to}\; \mathsf{Ne}^{B}$$

$$\mathsf{abs} \;:\; [A]\,\mathsf{Nf}^{B} \;\dot{\to}\; \mathsf{Nf}^{A \Rightarrow B}$$

$$\mathsf{zero} \;:\; 1 \;\dot{\to}\; [A]\,\mathsf{Var}^{A}$$

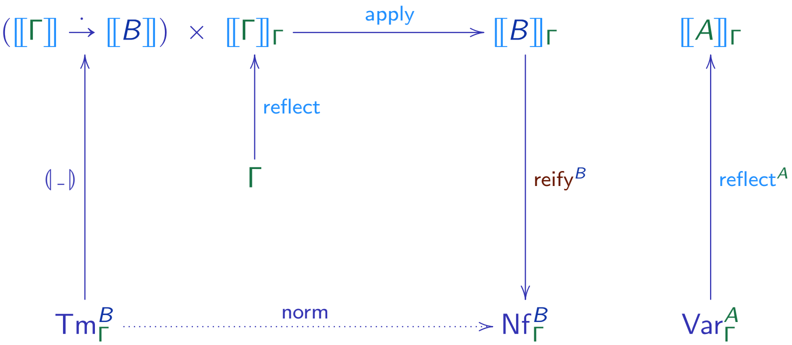
$$\mathsf{suc} \;:\; \mathsf{Var}^{A} \;\dot{\to}\; [A']\,\mathsf{Var}^{A}$$

Normal forms

M

$$\begin{array}{llll}
\mathsf{var} & : & \mathsf{Var}^A & \dot{\to} \; \mathsf{Ne}^A \\
\mathsf{app} & : & \mathsf{Ne}^{A \Rightarrow B} \,\dot{\times}\, \mathsf{Nf}^A & \dot{\to} \; \mathsf{Ne}^B \\
\mathsf{abs} & : & [A]\,\mathsf{Nf}^B & \dot{\to} \; \mathsf{Nf}^{A \Rightarrow B} \\
\mathsf{ne} & : & \mathsf{Ne}^B & \dot{\to} \; \mathsf{Nf}^B \\
\mathsf{zero} & : & 1 & \dot{\to} \; [A]\,\mathsf{Var}^A \\
\mathsf{suc} & : & \mathsf{Var}^A & \dot{\to} \; [A']\,\mathsf{Var}^A
\end{array}$$

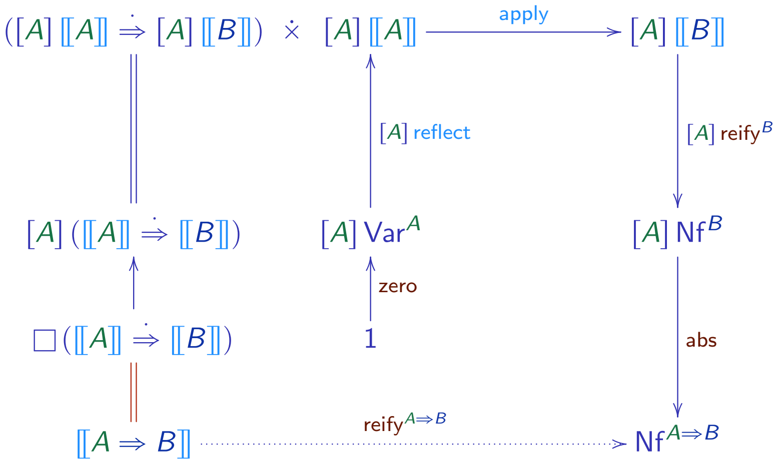
Normalization by evaluation



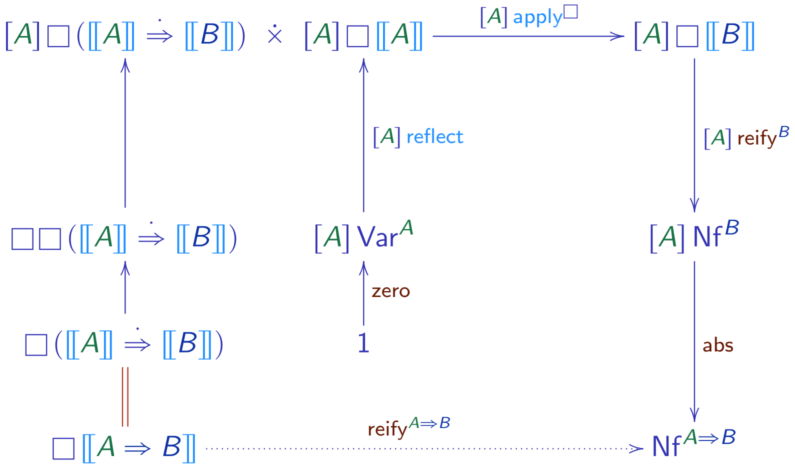
NbE type semantics

$$
\begin{array}{lcll}
[\![o]\!] & = & \mathsf{Nf}^{o} & \text{if } o \text{ is a base type} \\
[\![1]\!] & = & \dot{1} & \\
[\![A \times B]\!] & = & [\![A]\!] \mathbin{\dot\times} [\![B]\!] & \\
[\![A \Rightarrow B]\!] & = & \Box([\![A]\!] \mathbin{\dot\Rightarrow} [\![B]\!]) & \text{monotonization} \\
\mathsf{mon}^{A} & : & [\![A]\!] \mathbin{\dot\to} \Box [\![A]\!] & \text{monotonicity} \\
 & & & \text{“presheaf”}
\end{array}
$$

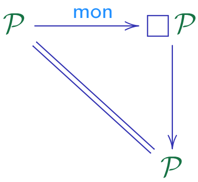
Reification $[\![B]\!] \dot{\to} \mathsf{Tm}^B$



Reification $\Box [\![B]\!] \mathrel{\dot\to} \mathrm{Tm}^B$



Monotone (positive) types



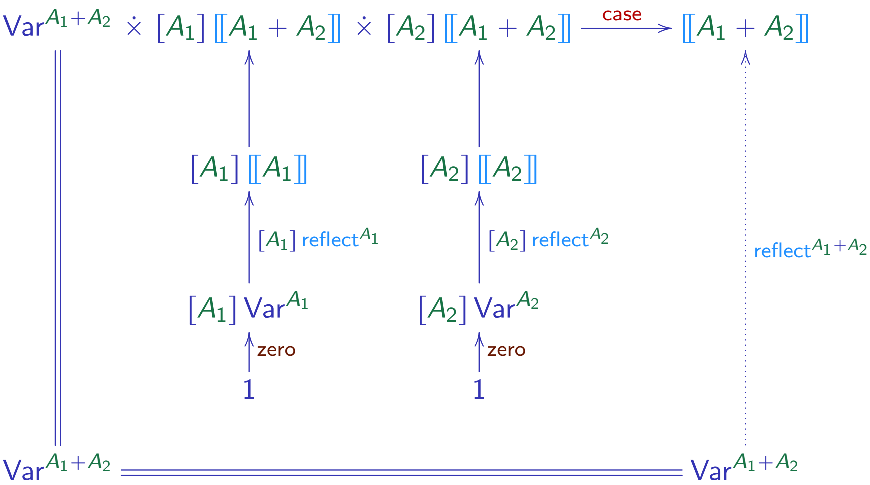
comonad coalgebra

$$\begin{aligned}\mathcal{P} \;::=\;& \Box\mathcal{N} \mid \dot{1} \mid \mathcal{P} \mathbin{\dot{\times}} \mathcal{P}' \mid \dot{0} \mid \mathcal{P} \mathbin{\dot{+}} \mathcal{P}' \\ \mid\;& \mathsf{Var}^A \mid \mathsf{Ne}^A \mid \mathsf{Nf}^A \mid \mathsf{Tm}^A\end{aligned}$$

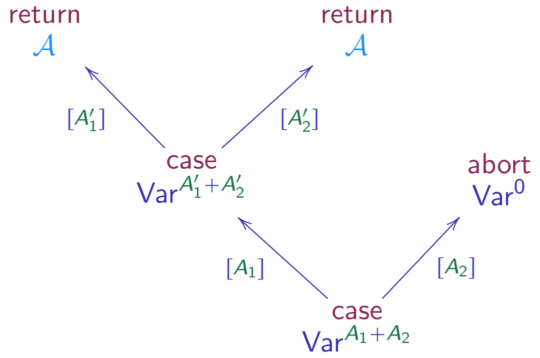
Sum types

$$
\begin{array}{lcll}
[\![0]\!] & = & \dot{0} & ? \\
[\![A_1 + A_2]\!] & = & [\![A_1]\!] \dotplus [\![A_2]\!] & ?
\end{array}
$$

Reflection $\mathsf{Var}^A \mathrel{\dot{\to}} [\![A]\!]$



Case trees $\Diamond\mathcal{A}$



Cover monad ♦ (∼AF)

$$\begin{array}{lll} \mathsf{abort} & : & \mathsf{Var}^0 \mathrel{\dot\to} \blacklozenge\mathcal{A} \\ \mathsf{case} & : & \mathsf{Var}^{A_1+A_2} \mathrel{\dot\times} ([A_1]\,\blacklozenge\mathcal{A}) \mathrel{\dot\times} ([A_2]\,\blacklozenge\mathcal{A}) \mathrel{\dot\to} \blacklozenge\mathcal{A} \end{array}$$

services

$$\begin{array}{lll} \mathsf{return} & : & \mathcal{A} \mathrel{\dot\to} \blacklozenge\mathcal{A} \\ \mathsf{join} & : & \blacklozenge\blacklozenge\mathcal{A} \mathrel{\dot\to} \blacklozenge\mathcal{A} \end{array}$$

monad

$$\mathsf{map} \quad : \quad (\mathcal{A} \mathrel{\dot\to} \mathcal{B}) \to (\blacklozenge\mathcal{A} \mathrel{\dot\to} \blacklozenge\mathcal{B})$$

functor

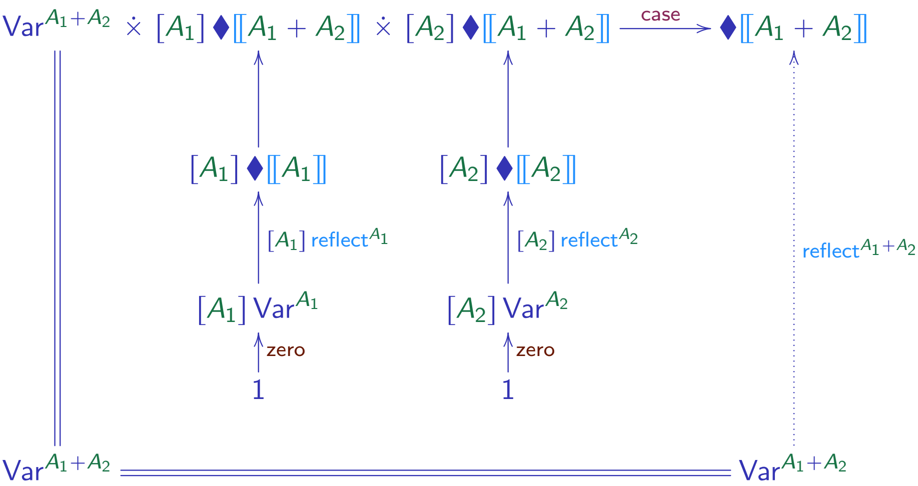
$$\widehat{\mathsf{map}} \quad : \quad \Box(\mathcal{A} \mathrel{\dot\Rightarrow} \mathcal{B}) \mathrel{\dot\times} \blacklozenge\mathcal{A} \mathrel{\dot\to} \blacklozenge\mathcal{B}$$

strong functor

$$\blacklozenge\Box\mathcal{A} \mathrel{\dot\to} \Box\blacklozenge\mathcal{A}$$

commutative law

Reflection $\mathrm{Var}^A \mathrel{\dot\to} \blacklozenge[\![A]\!]$

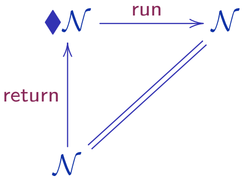


Sum types

$$
\begin{aligned}
[\![0]\!] &= \blacklozenge\dot{0} \\
[\![A_1 + A_2]\!] &= \blacklozenge([\![A_1]\!] \dotplus [\![A_2]\!])
\end{aligned}
$$

sufficient?

Monadic (runnable, negative) types



monad algebra

$$\begin{array}{lll} \mathcal{N} & ::= & \blacklozenge\mathcal{P} \mid \dot{1} \mid \mathcal{N} \mathbin{\dot{\times}} \mathcal{N}' \mid \mathcal{P} \mathbin{\dot{\Rightarrow}} \mathcal{N} \\ & \mid & \mathsf{Nf}^{A} \mid \mathsf{Tm}^{A} \end{array}$$

computation types

$$\mathcal{P} \quad ::= \quad \Box\mathcal{N} \mid \dot{1} \mid \mathcal{P} \mathbin{\dot{\times}} \mathcal{P}' \mid \dot{0} \mid \mathcal{P} \mathbin{\dot{+}} \mathcal{P}'$$

value types

NbE standard interpretation (sheaves, mon & run)

$$
\begin{array}{lcl}
[\![0]\!] & = & \blacklozenge\dot{0} \\
[\![A_1 + A_2]\!] & = & \blacklozenge([\![A_1]\!] \dotplus [\![A_2]\!]) \\
\\
[\![1]\!] & = & \dot{1} \\
[\![A \times B]\!] & = & [\![A]\!] \mathbin{\dot{\times}} [\![B]\!] \\
\\
[\![A \Rightarrow B]\!] & = & \square([\![A]\!] \mathrel{\dot{\Rightarrow}} [\![B]\!])
\end{array}
$$

$$
\begin{array}{lclcl}
\mathsf{reflect}^A & : & \mathsf{Ne}^A & \dot{\to} & [\![A]\!] \\
\mathsf{reify}^B & : & [\![B]\!] & \dot{\to} & \mathsf{Nf}^B
\end{array}
$$

Call-by-name interpretation (computation types)

$$
\begin{array}{lcl}
[\![0]\!]^- & = & \blacklozenge\dot{0} \\
[\![A_1 + A_2]\!]^- & = & \blacklozenge(\Box\,[\![A_1]\!]^- \dot{+} \Box\,[\![A_2]\!]^-)
\end{array}
$$

$$
\begin{array}{lcl}
[\![1]\!]^- & = & \dot{1} \\
[\![B_1 \times B_2]\!]^- & = & [\![B_1]\!]^- \dot{\times} [\![B_2]\!]^-
\end{array}
$$

$$
[\![A \Rightarrow B]\!]^- \quad = \quad \Box\,[\![A]\!]^- \dot{\Rightarrow} [\![B]\!]^-
$$

$$
\begin{array}{lcccl}
\mathsf{reflect}^N & : & \mathsf{Ne}^N & \dot{\rightarrow} & [\![N]\!]^- \\
\mathsf{reify}^N & : & \Box\,[\![N]\!]^- & \dot{\rightarrow} & \mathsf{Nf}^N
\end{array}
$$

Call-by-value interpretation (value types)

$$
\begin{array}{lcl}
[\![0]\!]^{+} & = & \dot{0} \\
[\![A_1 + A_2]\!]^{+} & = & [\![A_1]\!]^{+} \dot{+} [\![A_2]\!]^{+} \\
\\
[\![1]\!]^{+} & = & \dot{1} \\
[\![A_1 \times A_2]\!]^{+} & = & [\![A_1]\!]^{+} \dot{\times} [\![A_2]\!]^{+} \\
\\
[\![A \Rightarrow B]\!]^{+} & = & \Box([\![A]\!]^{+} \dot{\Rightarrow} \blacklozenge [\![B]\!]^{+})
\end{array}
$$

$$
\begin{array}{lcccl}
\mathsf{reflect}^{P} & : & \mathsf{Ne}^{P} & \dot{\to} & \blacklozenge [\![P]\!]^{+} \\
\mathsf{reify}^{P} & : & [\![P]\!]^{+} & \dot{\to} & \mathsf{Nf}^{P}
\end{array}
$$

Optimal interpretation (mixed, CBPV)

$$
\begin{aligned}
[\![0]\!]^+ &= \dot{0} & [\![0]\!]^- &= \blacklozenge[\![0]\!]^+ \\
[\![A_1 + A_2]\!]^+ &= [\![A_1]\!]^+ \dot{+} [\![A_2]\!]^+ & [\![A_1 + A_2]\!]^- &= \blacklozenge[\![A_1 + A_2]\!]^+ \\
[\![A \Rightarrow B]\!]^+ &= \square\,[\![A \Rightarrow B]\!]^- & [\![A \Rightarrow B]\!]^- &= [\![A]\!]^+ \dot{\Rightarrow} [\![B]\!]^-
\end{aligned}
$$

$$
\begin{aligned}
\mathsf{reflect}^P &: \mathsf{Ne}^P \dot{\to} \blacklozenge[\![P]\!]^+ & \mathsf{reflect}^N &: \mathsf{Ne}^N \dot{\to} [\![N]\!]^- \\
\mathsf{reify}^P &: [\![P]\!]^+ \dot{\to} \mathsf{Nf}^P & \mathsf{reify}^N &: \square\,[\![N]\!]^- \dot{\to} \mathsf{Nf}^N
\end{aligned}
$$

Cliffhanger?

Normalization by Evaluation for Call-by-Push-Value
and Polarized Lambda-Calculus

Andreas Abel, Christian Sattler
https://arxiv.org/abs/1902.06097

Making Of

Agda code

https://github.com/andreasabel/ipl/

Credits

- **Catarina Coquand**: From Semantics to Rules: A Machine Assisted Analysis. CSL 1993.
- **Achim Jung, Jerzy Tiuryn**: A New Characterization of Lambda Definability. TLCA 1993.
- **Thorsten Altenkirch, Martin Hofmann, Thomas Streicher**: Categorical Reconstruction of a Reduction Free Normalization Proof. CTCS 1995.
- **Olivier Danvy**: Type-Directed Partial Evaluation. POPL 1996.
- **T. Altenkirch, P. Dybjer, M. Hofmann, Ph. Scott**: Normalization by Evaluation for Typed Lambda Calculus with Coproducts. LICS 2001.
- **Paul Blain Levy**: Call-by-push-value: Decomposing call-by-value and call-by-name. HOSC 19(4), 2006.
- **Freiric Barral**: Decidability for non-standard conversions in λ-calculus. PhD thesis, LMU Munich, 2008.
- **Gabriel Scherer**: Deciding equivalence with sums and the empty type. POPL 2017.
- **G. Allais, R. Atkey, J. Chapman, C. McBride, J. McKinna**: A type and scope safe universe of syntaxes with binding. ICFP 2018.