Algebraic Foundations for Effect-Dependent Optimisations

Ohad Kammar   Gordon Plotkin

Principles of Programming Seminar
Carnegie Mellon University
January 30, 2012

POPL'12

Laboratory for Foundations of Computer Science
Association for Computing Machinery
sicsa*
The Royal Society
The University of Edinburgh informatics

* Gifford-style types and effects

Effect systems

$$\ell_1 := 1;$$
$$\ell_2 := \mathbf{deref}(\ell_3)$$

* Gifford-style types and effects

Effect systems

$$\begin{aligned}\vdash \ell_1 &:= \mathtt{1};\\ \ell_2 &:= \mathbf{deref}(\ell_3) : () \,!\, \underbrace{\{\mathtt{lookup}, \mathtt{update}\}}_{\varepsilon}\end{aligned}$$

$$\Gamma \vdash M : A \,!\, \varepsilon$$

Effect-dependent optimisations [Benton et al.]

Swap:
$$\begin{array}{c} \vdash M_i : () \,!\, \varepsilon_i, \\ \varepsilon_i \subseteq \{\mathtt{lookup}\} \end{array} \implies \begin{array}{c} M_1; M_2; N \\ = \\ M_2; M_1; N \end{array}$$

A language a paper

- N. Benton and A. Kennedy. *Monads, effects and transformations*, 1999.
- N. Benton, A. Kennedy, L. Beringer, M. Hofmann. *Reading, writing and relations*, 2006.
- N. Benton and P. Buchlovsky. *Semantics of an effect analysis for exceptions*, 2007.
- N. Benton, A. Kennedy, L. Beringer, M. Hofmann. *Relational semantics for effect-based program transformations with dynamic allocation*, 2007.
- N. Benton, A. Kennedy, L. Beringer, M. Hofmann. *Relational semantics for effect-based program transformations: higher-order store*, 2009.
- J. Thamsborg, L. Birkedal. *A kripke logical relation for effect-based program transformations*, 2011.

Contribution

Craft

case by case treatment

$\Downarrow$

Science

general semantic account of Gifford-style effect type systems

$\Downarrow$

Engineering

- results: validate optimisations that occur in practice
- tools: to assist validation and instrumentation, e.g. optimisation tables
- methods: for overcoming difficulties, e.g. equational reasoning for modular validation

Observation [Wadler'98]

Change notation:

$$\Gamma \vdash M : A \,!\, \varepsilon \implies \Gamma \vdash M : T_\varepsilon A$$

$T_\varepsilon A$ is an indexed family of monadic types.

$$\begin{tikzcd}[column sep=small, row sep=small] & \varepsilon_4 & \\ \varepsilon_2 \arrow[ur, phantom, sloped, "\subseteq"] & & \varepsilon_3 \arrow[ul, phantom, sloped, "\supseteq"] \\ & \varepsilon_1 \arrow[ul, phantom, sloped, "\supseteq"] \arrow[ur, phantom, sloped, "\subseteq"] & \end{tikzcd} \implies \begin{tikzcd}[column sep=small, row sep=small] & T_{\varepsilon_4} & \\ T_{\varepsilon_2} \arrow[ur] & & T_{\varepsilon_3} \arrow[ul] \\ & T_{\varepsilon_1} \arrow[ul] \arrow[ur] & \end{tikzcd}$$

effect inclusions   monad morphisms

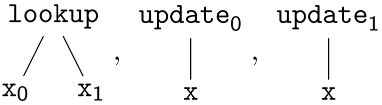
Suggested monads for global state

$$
\begin{tikzcd}[column sep=tiny]
& T_{\{\texttt{lookup},\texttt{update}\}}(A) = (2 \times A)^2 & \\
T_{\{\texttt{lookup}\}}(A) = A^2 \arrow[ur, hook] & & T_{\{\texttt{update}\}}(A) = (1+2) \times A \arrow[ul, hook'] \\
& T_{\emptyset}(A) = A \arrow[ul, hook'] \arrow[ur, hook] &
\end{tikzcd}
$$

* Algebraic theory of effects [Plotkin and Power]

An interface to effects:

Effect operations Σ e.g.: $\texttt{lookup} : 2$, $\texttt{update} : 1\ \langle 2 \rangle$



* Algebraic theory of effects [Plotkin and Power]

An interface to effects:

Effect operations Σ e.g.: $\texttt{lookup} : 2$, $\texttt{update} : 1\ \langle 2 \rangle$

Effect equations E e.g.:

$$
\begin{tikzcd}[column sep=small]
\texttt{update}_0 \arrow[d, dash] & & \\
\texttt{update}_1 \arrow[d, dash] & = & \texttt{update}_1 \arrow[d, dash] \\
\texttt{x} & & \texttt{x}
\end{tikzcd}
\qquad
\begin{tikzcd}[column sep=tiny]
 & & & \texttt{lookup} \arrow[dll, dash] \arrow[drr, dash] & & & & & & \\
 & \texttt{lookup} \arrow[dl, dash] \arrow[dr, dash] & & & & \texttt{lookup} \arrow[dl, dash] \arrow[dr, dash] & & = & & \texttt{lookup} \arrow[dl, dash] \arrow[dr, dash] & \\
\texttt{x}_{00} & & \texttt{x}_{01} & & \texttt{x}_{10} & & \texttt{x}_{11} & & \texttt{x}_{00} & & \texttt{x}_{11}
\end{tikzcd}
$$

Each theory $\langle \Sigma, E \rangle$ **generates** a monad T (free model).

Algebraic view

Key observation

ε as an algebraic **signature**.

Global state

For $\Sigma := \{\mathtt{lookup} : 2, \mathtt{update} : 1\,\langle 2\rangle\}$,

$$\varepsilon = \emptyset, \{\mathtt{lookup}\}, \{\mathtt{update}\}, \{\mathtt{lookup}, \mathtt{update}\}$$

A **novel** banality.

* Conservative restriction

Global state

$$E :=$$

$$\mathrm{Th}\left\{ \begin{array}{c}
\begin{array}{c}\mathtt{update}_b\\ | \\ \mathtt{update}_{b'}\\ | \\ \mathtt{x}\end{array} = \begin{array}{c}\mathtt{update}_{b'}\\ | \\ \mathtt{x}\end{array}, \\[1em]
\begin{array}{c}\mathtt{update}_b\\ | \\ \begin{array}{ccc} & \mathtt{lookup} & \\ / & & \backslash \\ \mathtt{x}_0 & & \mathtt{x}_1\end{array}\end{array} = \begin{array}{c}\mathtt{update}_b\\ | \\ \mathtt{x}_b\end{array}, \\[1em]
\begin{array}{ccc} & \mathtt{lookup} & \\ / & & \backslash \\ \mathtt{update}_0 & & \mathtt{update}_1 \\ | & & | \\ \mathtt{x} & & \mathtt{x}\end{array} = \mathtt{x}
\end{array} \right\}$$

* Conservative restriction

Global state

$E_\varepsilon = \{ s = t \in E \,|\, s,\ t \text{ are } \varepsilon\text{-terms} \}$

$$
\begin{array}{ccccc}
 & & E_{\{\mathtt{lookup},\mathtt{update}\}} & & \\
 & \rotatebox{-45}{$\supseteq$} & & \rotatebox{45}{$\supseteq$} & \\
E_{\{\mathtt{lookup}\}} & & & & E_{\{\mathtt{update}\}} \\
 & \rotatebox{45}{$\supseteq$} & & \rotatebox{-45}{$\supseteq$} & \\
 & & E_{\emptyset} & &
\end{array}
$$

$$
E_{\{\mathtt{lookup},\mathtt{update}\}} =
$$

$$
\mathrm{Th}\left\{
\begin{array}{c}
\begin{array}{c}\mathtt{update}_b\\ |\\ \mathtt{update}_{b'}\\ |\\ \mathtt{x}\end{array}
=
\begin{array}{c}\mathtt{update}_{b'}\\ |\\ \mathtt{x}\end{array}
,\\[2em]
\begin{array}{c}\mathtt{update}_b\\ |\\ \mathtt{lookup}\\ / \quad \backslash\\ \mathtt{x}_0 \quad \mathtt{x}_1\end{array}
=
\begin{array}{c}\mathtt{update}_b\\ |\\ \mathtt{x}_b\end{array}
,\\[2em]
\begin{array}{c}\mathtt{lookup}\\ / \qquad\qquad \backslash\\ \begin{array}{c}\mathtt{update}_0\\ |\\ \mathtt{x}\end{array} \quad \begin{array}{c}\mathtt{update}_1\\ |\\ \mathtt{x}\end{array}\end{array}
= \mathtt{x}
\end{array}
\right\}
$$

* Conservative restriction

Global state

$E_\varepsilon = \{s = t \in E \mid s,\ t \text{ are } \varepsilon\text{-terms}\}$

$$
\begin{tikzcd}[column sep=small]
 & E_{\{\mathtt{lookup},\mathtt{update}\}} & \\
E_{\{\mathtt{lookup}\}} \arrow[ur, phantom, "\subseteq" sloped] & & E_{\{\mathtt{update}\}} \arrow[ul, phantom, "\supseteq" sloped] \\
 & E_{\emptyset} \arrow[ul, phantom, "\supseteq" sloped] \arrow[ur, phantom, "\subseteq" sloped] &
\end{tikzcd}
$$

$$
E_{\{\mathtt{lookup}\}} = \mathrm{Th}\left\{
\begin{array}{c}
\mathtt{lookup}(\mathtt{lookup}(\mathtt{x}_{00}, \mathtt{x}_{01}), \mathtt{lookup}(\mathtt{x}_{10}, \mathtt{x}_{11})) = \mathtt{lookup}(\mathtt{x}_{00}, \mathtt{x}_{11}), \\
\mathtt{lookup}(\mathtt{x}, \mathtt{x}) = \mathtt{x}
\end{array}
\right\}
$$

* Conservative restriction

Global state

$E_\varepsilon = \{s = t \in E \mid s,\ t \text{ are } \varepsilon\text{-terms}\}$

$$
\begin{tikzcd}
& E_{\{\mathtt{lookup},\mathtt{update}\}} & \\
E_{\{\mathtt{lookup}\}} \arrow[ur, phantom, sloped, "\subseteq"] & & E_{\{\mathtt{update}\}} \arrow[ul, phantom, sloped, "\supseteq"] \\
& E_{\emptyset} \arrow[ul, phantom, sloped, "\subseteq"] \arrow[ur, phantom, sloped, "\supseteq"] &
\end{tikzcd}
$$

$$E_{\{\mathtt{lookup}\}} = \mathrm{Th}\left\{ \begin{array}{c} \mathtt{lookup}(\mathtt{lookup}(\mathtt{x}_{00}, \mathtt{x}_{01}), \mathtt{lookup}(\mathtt{x}_{10}, \mathtt{x}_{11})) = \mathtt{lookup}(\mathtt{x}_{00}, \mathtt{x}_{11}), \\ \mathtt{lookup}(\mathtt{x}, \mathtt{x}) = \mathtt{x} \end{array} \right\}$$

Compare:

$$E = \mathrm{Th}\left\{ \mathtt{update}_b(\mathtt{update}_{b'}(\mathtt{x})) = \mathtt{update}_{b'}(\mathtt{x}),\ \mathtt{update}_b(\mathtt{lookup}(\mathtt{x}_0, \mathtt{x}_1)) = \mathtt{update}_b(\mathtt{x}_b),\ \mathtt{lookup}(\mathtt{update}_0(\mathtt{x}), \mathtt{update}_1(\mathtt{x})) = \mathtt{x} \right\}$$

* Conservative restriction

Global state

$E_\varepsilon = \{s = t \in E \,|\, s,\ t \text{ are } \varepsilon\text{-terms}\}$

$$\begin{tikzcd}[column sep=small]
 & E_{\{\mathtt{lookup},\mathtt{update}\}} & \\
E_{\{\mathtt{lookup}\}} \arrow[ur, phantom, "\subseteq" sloped] & & \color{red}{E_{\{\mathtt{update}\}}} \arrow[ul, phantom, "\supseteq" sloped] \\
 & E_{\emptyset} \arrow[ul, phantom, "\supseteq" sloped] \arrow[ur, phantom, "\subseteq" sloped] &
\end{tikzcd}$$

$$E_{\{\mathtt{update}\}} = \mathrm{Th}\left\{ \mathtt{update}_b(\mathtt{update}_{b'}(\mathtt{x})) = \mathtt{update}_{b'}(\mathtt{x}) \right\}$$

* Conservative restriction

Global state

$E_\varepsilon = \{s = t \in E \,|\, s,\ t \text{ are } \varepsilon\text{-terms}\}$

$$\begin{array}{ccc} & E_{\{\texttt{lookup},\texttt{update}\}} & \\ \rotatebox{45}{$\subseteq$} & & \rotatebox{-45}{$\supseteq$} \\ E_{\{\texttt{lookup}\}} & & E_{\{\texttt{update}\}} \\ \rotatebox{-45}{$\supseteq$} & & \rotatebox{45}{$\subseteq$} \\ & E_\emptyset & \end{array}$$

$E_\emptyset = \mathrm{Th}\,\emptyset$

* Conservative restriction

Derived monads

$$
\begin{tikzcd}
 & T_{\{\mathtt{lookup},\mathtt{update}\}}(A) = (2 \times A)^2 & \\
T_{\{\mathtt{lookup}\}}(A) = A^2 \arrow[ur, hook] & & T_{\{\mathtt{update}\}}(A) = (1 + 2) \times A \arrow[ul, hook'] \\
 & T_{\emptyset}(A) = A \arrow[ul, hook'] \arrow[ur, hook] &
\end{tikzcd}
$$

Structure

- Optimisations
  - structural, algebraic (local), abstract (global)
  - unification
  - discovery
- Conclusion

Optimisations

Structural properties

Valid for all T_ε

e.g.

- β, η rules
- sequencing

$$(M; N); P = M; (N; P)$$

Practically

Bread and butter of optimisation, e.g.

- constant propagation
- common subexpression elimination
- loop unrolling

etc..

Local algebraic properties

Single equations in E_ε, e.g.:

$$
\begin{tikzcd}[column sep=small]
& \mathtt{update}_b \arrow[d, dash] & & \mathtt{update}_b \arrow[dd, dash] \\
& \mathtt{lookup} \arrow[dl, dash] \arrow[dr, dash] & = & \\
\mathtt{x}_0 & & \mathtt{x}_1 & \mathtt{x}_b
\end{tikzcd}
$$

become optimisations, e.g.:

$$
\begin{array}{lcl}
\ell := V; & & \ell := V; \\
\mathrm{y} \leftarrow \mathtt{deref}(\ell); & = & \\
N & & N[V/\mathrm{y}]
\end{array}
$$

note quantification over variables only (**local** property).

* Global algebraic properties

Algebraic characterisation

For all $t(\mathtt{x}_1, \ldots, \mathtt{x}_n)$:

$$\begin{array}{ccc} & t & \\ \diagup & & \diagdown \\ \mathtt{x} & \cdots & \mathtt{x} \end{array} = \mathtt{x}$$

note quantification over **terms** too (**global** property).

Discard

$$M; \mathtt{return}\,() = \mathtt{return}\,()$$

Knowledge unification

Discard	B F	$\dfrac{\Gamma \vdash_{\varepsilon} M : \mathbf{F}_{\varepsilon} A \quad \Gamma \vdash_{\varepsilon'} N : \underline{B}}{(\mathbf{coerce}\, M)\ \mathbf{to}\ x : A.\, N = N}$	B F	$\dfrac{\Gamma \vdash_{\varepsilon} M : \mathbf{F}_{\varepsilon} A}{M\ \mathbf{to}\ x : A.\, \mathbf{return}_{\varepsilon} \star = \mathbf{return}_{\varepsilon} \star}$

	F	$\mathcal{T}_{\varepsilon}$ affine: $$\eta^{\varepsilon}_{\mathbb{1}} : \mathbb{1} \to \lvert F_{\varepsilon} \mathbb{1} \rvert$$ has a continuous inverse	For all ε-terms t: $$t(\mathbf{x}, \ldots, \mathbf{x}) = \mathbf{x}$$

Knowledge unification

name		utilitarian form		pristine form		abstract side condition	algebraic equivalent	example basic theories
Discard	B F	$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A \quad \Gamma \vdash_{\varepsilon'} N : \underline{B}}{(\mathbf{coerce} M)\ \mathbf{to}\ x : A.\, N = N}$	B F	$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A}{M\ \mathbf{to}\ x : A.\, \mathbf{return}_\varepsilon \star = \mathbf{return}_\varepsilon \star}$	F	$\mathcal{T}_\varepsilon$ affine: $\eta^\varepsilon_{\mathbb{1}} : \mathbb{1} \to \lvert F_\varepsilon \mathbb{1} \rvert$ has a continuous inverse	For all ε-terms t: $t(\mathbf{x}, \dots, \mathbf{x}) = \mathbf{x}$	read-only state, convex, upper and lower semilattices
Copy	B F	$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A \quad \Gamma, x : A, y : A \vdash_{\varepsilon'} N : \underline{B}}{\begin{array}{l}\mathbf{coerce} M\ \mathbf{to}\ x : A.\\ \mathbf{coerce} M\ \mathbf{to}\ y : A.\, N =\\ \mathbf{coerce} M\ \mathbf{to}\ x : A.\, N[x/y]\end{array}}$	B F	$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A}{\begin{array}{c}M\ \mathbf{to}\ x : A.\, M\ \mathbf{to}\ y : A.\, \mathbf{return}_\varepsilon(x, y)\\ =\\ M\ \mathbf{to}\ x : A.\, \mathbf{return}_\varepsilon(x, x)\end{array}}$	F	$\mathcal{T}_\varepsilon$ relevant: $\psi_\varepsilon \circ \delta = L^\varepsilon \delta$	For all ε-terms t: $t(t(\mathbf{x}_{11}, \dots, \mathbf{x}_{1n}), \dots, t(\mathbf{x}_{n1}, \dots, \mathbf{x}_{nn})) = t(\mathbf{x}_{11}, \dots, \mathbf{x}_{nn})$	exceptions, lifting, read-only state, write-only state
Weak Copy		$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A \quad \Gamma, y : A \vdash_{\varepsilon'} N : \underline{B}}{\begin{array}{l}\mathbf{coerce} M\ \mathbf{to}\ x : A.\\ \mathbf{coerce} M\ \mathbf{to}\ y : A.\, N =\\ \mathbf{coerce} M\ \mathbf{to}\ y : A.\, N\end{array}}$		$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A}{M\ \mathbf{to}\ x : A.\, M = M}$		$\mu^\varepsilon \circ L^\varepsilon \pi_1 \circ \mathrm{str}^\varepsilon \circ \delta = \mathrm{id}$	For all ε-terms t: $t(t(\mathbf{x}_1, \dots, \mathbf{x}_n), \dots, t(\mathbf{x}_1, \dots, \mathbf{x}_n)) = t(\mathbf{x}_1, \dots, \mathbf{x}_n)$	any affine or relevant theory: lifting, exceptions, read-only and write-only state, all three semilattice theories
Swap	B F	$\dfrac{\Gamma \vdash_{\varepsilon_1} M_1 : \mathbf{F}_{\varepsilon_1} A_1 \quad \Gamma \vdash_{\varepsilon_2} M_2 : \mathbf{F}_{\varepsilon_2} A_2 \quad \Gamma, x_1 : A_1, x_2 : A_2 \vdash_{\varepsilon'} N}{\begin{array}{l}\mathbf{coerce} M_1\ \mathbf{to}\ x_1 : A_1.\\ \mathbf{coerce} M_2\ \mathbf{to}\ x_2 : A_2.\, N =\\ \mathbf{coerce} M_2\ \mathbf{to}\ x_2 : A_2.\\ \mathbf{coerce} M_1\ \mathbf{to}\ x_1 : A_1.\, N\end{array}}$	B F	$\dfrac{\Gamma \vdash_{\varepsilon_1} M_1 : \mathbf{F}_{\varepsilon_1} A_1 \quad \Gamma \vdash_{\varepsilon_2} M_2 : \mathbf{F}_{\varepsilon_2} A_2}{\begin{array}{l}\mathbf{coerce} M_1\ \mathbf{to}\ x_1 : A_1.\\ \mathbf{coerce} M_2\ \mathbf{to}\ x_2 : A_2.\, \mathbf{return}_\varepsilon\,(x_1, x_2) =\\ \mathbf{coerce} M_2\ \mathbf{to}\ x_2 : A_2.\\ \mathbf{coerce} M_1\ \mathbf{to}\ x_1 : A_1.\, \mathbf{return}_\varepsilon\,(x_1, x_2)\end{array}}$	F	$\mathfrak{T}_{\varepsilon_1 \subseteq \varepsilon}, \mathfrak{T}_{\varepsilon_2 \subseteq \varepsilon}$ commute: $\psi_\varepsilon \circ (m^{\varepsilon_1 \subseteq \varepsilon} \times m^{\varepsilon_2 \subseteq \varepsilon}) = \tilde{\psi}_\varepsilon \circ (m^{\varepsilon_1 \subseteq \varepsilon} \times m^{\varepsilon_2 \subseteq \varepsilon})$	$\mathfrak{T}_{\varepsilon_1 \subseteq \varepsilon}$ translations commute with $\mathfrak{T}_{\varepsilon_2 \subseteq \varepsilon}$ translations (see tensor equations)	$\mathcal{T}_1 \to \mathcal{T}_1 \otimes \mathcal{T}_2 \leftarrow \mathcal{T}_2$, e.g., distinct global memory cells
Weak Swap		$\dfrac{\Gamma \vdash_{\varepsilon_1} M_1 : \mathbf{F}_{\varepsilon_1} A_1 \quad \Gamma \vdash_{\varepsilon_2} M_2 : \mathbf{F}_{\varepsilon_2} A_2 \quad \Gamma, x_1 : A_1 \vdash_{\varepsilon'} N}{\text{(same as Swap)}}$		$\dfrac{\Gamma \vdash_{\varepsilon_1} M_1 : \mathbf{F}_{\varepsilon_1} A_1 \quad \Gamma \vdash_{\varepsilon_2} M_2 : \mathbf{F}_{\varepsilon_2} A_2}{\begin{array}{l}\mathbf{coerce} M_1\ \mathbf{to}\ x_1 : A_1.\\ \mathbf{coerce} M_2\ \mathbf{to}\ x_2 : A_2.\, \mathbf{return}_\varepsilon x_1 =\\ \mathbf{coerce} M_2\ \mathbf{to}\ x_2 : A_2.\\ \mathbf{coerce} M_1\ \mathbf{to}\ x_1 : A_1.\, \mathbf{return}_\varepsilon x_1\end{array}}$		$\psi_\varepsilon \circ (m^{\varepsilon_1} \times m^{\varepsilon_2}) \circ (\mathrm{id} \times \eta^{\varepsilon_2}_{\mathbb{1}}) = \tilde{\psi}_\varepsilon \circ (m^{\varepsilon_1} \times m^{\varepsilon_2}) \circ (\mathrm{id} \times \eta^{\varepsilon_2}_{\mathbb{1}})$	For all ε-terms $t = \mathfrak{T}_1(t')$, $s = \mathfrak{T}_2(s')$: $t(s(\mathbf{x}_1, \dots, \mathbf{x}_1), \dots, s(\mathbf{x}_n, \dots, \mathbf{x}_n)) = s(t(\mathbf{x}_1, \dots, \mathbf{x}_n), \dots, t(\mathbf{x}_1, \dots, \mathbf{x}_n))$	when $\mathcal{T}_{\varepsilon_2}$ is affine, e.g.: read-only state and convex, upper and lower semilattices.
Isolated Swap		$\dfrac{\Gamma \vdash_{\varepsilon_1} M_1 : \mathbf{F}_{\varepsilon_1} A_1 \quad \Gamma \vdash_{\varepsilon_2} M_2 : \mathbf{F}_{\varepsilon_2} A_2 \quad \Gamma \vdash_{\varepsilon'} N}{\text{(same as Swap)}}$		$\dfrac{\Gamma \vdash_{\varepsilon_1} M_1 : \mathbf{F}_{\varepsilon_1} A_1 \quad \Gamma \vdash_{\varepsilon_2} M_2 : \mathbf{F}_{\varepsilon_2} A_2}{\begin{array}{l}\mathbf{coerce} M_1\ \mathbf{to}\ x_1 : A_1.\\ \mathbf{coerce} M_2\ \mathbf{to}\ x_2 : A_2.\, \mathbf{return}_\varepsilon \star =\\ \mathbf{coerce} M_2\ \mathbf{to}\ x_2 : A_2.\\ \mathbf{coerce} M_1\ \mathbf{to}\ x_1 : A_1.\, \mathbf{return}_\varepsilon \star\end{array}}$		$\psi_\varepsilon \circ (m^{\varepsilon_1} \times m^{\varepsilon_2}) \circ (\eta^{\varepsilon_1}_{\mathbb{1}} \times \eta^{\varepsilon_2}_{\mathbb{1}}) = \tilde{\psi}_\varepsilon \circ (m^{\varepsilon_1} \times m^{\varepsilon_2}) \circ (\eta^{\varepsilon_1}_{\mathbb{1}} \times \eta^{\varepsilon_2}_{\mathbb{1}})$	For all ε-terms $t = \mathfrak{T}_1(t')$, $s = \mathfrak{T}_2(s')$: $t(s(\mathbf{x}, \dots, \mathbf{x}), \dots, s(\mathbf{x}, \dots, \mathbf{x})) = s(t(\mathbf{x}, \dots, \mathbf{x}), \dots, t(\mathbf{x}, \dots, \mathbf{x}))$	when $\mathcal{T}\varepsilon_1$ is affine: read-only state and convex, upper and lower semilattices.
Unique	B	$\dfrac{\Gamma \vdash_\varepsilon M_i : \mathbf{F}_\varepsilon \mathbf{0}, i = 1, 2}{M_1 = M_2}$		(same as utilitarian form)		$F_\varepsilon \mathbb{0} = \mathbb{0}, \mathbb{1}$	$\mathcal{T}_\varepsilon$ equates all ε-constants	all three state theories, all three semilattice theories, a single unparameterised exception, lifting
Pure Hoist	B F	$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A \quad \Gamma, x : A \vdash_{\varepsilon'} N : \underline{B}}{\begin{array}{c}\mathbf{return}_\varepsilon\ \mathbf{thunk}\ (\mathbf{coerce} M\ \mathbf{to}\ x : A.\, N)\\ = M\ \mathbf{to}\ x : A.\, \mathbf{return}_\varepsilon\ \mathbf{thunk}\ N\end{array}}$	B F	$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A}{\begin{array}{l}\mathbf{return}_\varepsilon\ \mathbf{thunk}\ M =\\ M\ \mathbf{to}\ x : A.\, \mathbf{return}_\varepsilon\ \mathbf{thunk}\ \mathbf{return}_\varepsilon x\end{array}}$	F	$L^\varepsilon \eta^\varepsilon_W = \eta^\varepsilon_{\lvert F_\varepsilon W \rvert}$	all ε-terms are equal to variables in $\mathcal{T}_\varepsilon$	the empty theory, inconsistent theories
Hoist	B F	$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A \quad \Gamma, x : A \vdash_{\varepsilon'} N : \underline{B}}{\begin{array}{l}M\ \mathbf{to}\ x : A.\\ \mathbf{return}_\varepsilon\ \mathbf{thunk}\ (\mathbf{coerce} M\ \mathbf{to}\ x : A.\, N)\\ = M\ \mathbf{to}\ x : A.\, \mathbf{return}_\varepsilon\ \mathbf{thunk}\ N\end{array}}$	B F	$\dfrac{\Gamma \vdash_\varepsilon M : \mathbf{F}_\varepsilon A}{\begin{array}{l}M\ \mathbf{to}\ x : A.\\ \quad \mathbf{thunk}\ \mathbf{return}_\varepsilon\ (x,\ \mathbf{thunk}\ M) =\\ M\ \mathbf{to}\ x : A.\\ \quad \mathbf{thunk}\ \mathbf{return}_\varepsilon\ (x,\ \mathbf{thunk}\ \mathbf{return}_\varepsilon x)\end{array}}$	F	$L^\varepsilon \langle \eta^\varepsilon, \mathrm{id} \rangle = \mathrm{str}^\varepsilon \circ \delta$	all ε-terms are either a variable or independent of their variables via $\mathcal{T}_\varepsilon$	all theories containing only constants: lifting and exceptions

Figure 7. Abstract Optimisations

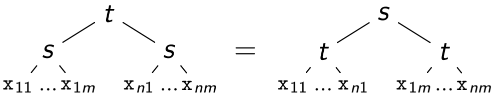
9

2012/1/20

* Global algebraic properties

Algebraic characterisation

For all ε_1-term $t(\mathtt{x}_1, \ldots, \mathtt{x}_n)$, and ε_2-term $s(\mathtt{x}_1, \ldots, \mathtt{x}_m)$:



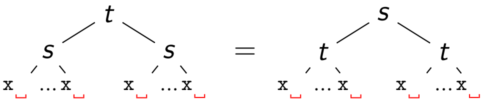
Swap

$$\mathtt{x} \leftarrow M_1; \mathtt{y} \leftarrow M_2; \mathtt{return}\ \langle \mathtt{x}, \mathtt{y} \rangle$$
$$=$$
$$\mathtt{y} \leftarrow M_2; \mathtt{x} \leftarrow M_1; \mathtt{return}\ \langle \mathtt{x}, \mathtt{y} \rangle$$

* Global algebraic properties

Algebraic characterisation

For all ε_1-term $t(\mathtt{x}_1, \ldots, \mathtt{x}_n)$, and ε_2-term $s(\mathtt{x}_1, \ldots, \mathtt{x}_m)$:



Isolated swap

$$M; N = N; M$$

Applicable for more effects.

Additional contributions

Details in the paper, and:

- An extended **example**:

$$\begin{aligned}\text{Exceptions} + (\text{Read Only} \otimes \text{Write Only} \otimes \text{Read-Write} \otimes \\ (\text{Rollback Exceptions} + \text{Input} + \text{Output}+ \\ (\text{Non-determinism} \otimes \text{Lifting})))\end{aligned}$$

  ($2^9 = 512$ effect sets).
- **Modular validation** of optimisations.
- Optimisation tables.

Caveats

- No effect inference.
- Not a rich logic (equational only).
- Only Gifford-style effect systems.
- Only algebraic effects.
- Did not cover all optimisations.

Summary

- N. Benton and A. Kennedy. *Monads, effects and transformations*, 1999.
- N. Benton, A. Kennedy, L. Beringer, M. Hofmann. *Reading, writing and relations*, 2006.
- N. Benton and P. Buchlovsky. *Semantics of an effect analysis for exceptions*, 2007.
- N. Benton, A. Kennedy, L. Beringer, M. Hofmann. *Relational semantics for effect-based program transformations with dynamic allocation*, 2007.
- N. Benton, A. Kennedy, L. Beringer, M. Hofmann. *Relational semantics for effect-based program transformations: higher-order store*, 2009.
- J. Thamsborg, L. Birkedal. *A kripke logical relation for effect-based program transformations*, 2011.

Summary

- **Category theory** was crucial to this formulation.
- The categorical characterisations connected to Führmann, Jacobs, Kock and Wraith.

Contribution

Craft

case by case treatment

$\Downarrow$

Science

general semantic account of Gifford-style effect type systems

$\Downarrow$

Engineering

- results: validate optimisations that occur in practice
- tools: to assist validation and instrumentation, e.g. optimisation tables
- methods: for overcoming difficulties, e.g. equational reasoning for modular validation

Appendices

- Atkey
- Further work

Abstract optimisations

(contd.) Discard: $\mathtt{x} \leftarrow M; \mathtt{return}_\varepsilon\, 0 = \mathtt{return}_\varepsilon\, 0$

Discard: Pristine Form

$$\frac{\Gamma \vdash M : T_\varepsilon A}{\mathtt{x} \leftarrow M; \mathtt{return}_\varepsilon\, 0 = \mathtt{return}_\varepsilon\, 0}$$

(cont.)

Categorical Characterisation

$$T_\varepsilon 1 \cong 1$$

Due to Kock, Jacobs, Führmann

Further work

- Effect reconstruction
- Handlers
- Automation
- More effects
- Locality

- Concurrency
- DSL reasoning.
- Richer program logics (Hoare, modal, etc.).

Isolated swap applicability

For example, if $\varepsilon_1 = \{\mathtt{input}\}, \varepsilon_2 = \{\mathtt{lookup}, \mathtt{update}\}$.

Atkey

Precise relationship of semantics is further work.
Similarities:

- Soundness of optimisations.
- Validation of the Benton et. al global state optimisations.
- Constructing a semantics out of an equational theory.

Differences:

- Our work included a general treatment of optimisations.
- Our work is tightly coupled to the algebraic semantics.
- Out work treats modular combinations of optimisations.

Perhaps our work can be generalised to the parametrised setting.