Graded Call-by-Push-Value

Tracking (co)effects in a functional language

Andreas Abel

Department of Computer Science and Engineering
Chalmers and University of Gothenburg, Sweden

LFCS seminar, U Edinburgh, Scotland
17 June 2025

Call-by-Push-Value (CBPV)

- Value (positive) types P

o	base type (e.g. integers)
$\Sigma_{i:I} P_i$	variants/sums with labels $i : I$
$\bigotimes_{i:I} P_i$	tuples/vectors with indices $i : I$
$\Box N$	thunks

- Computation (negative) types N

$\diamond P$	monadic type
$P \Rightarrow N$	function type (cbv)
$\Pi_{i:I} N_i$	records with fields $i : I$

CBPV typing: values

- A derivation of $\Gamma \vdash P$ is a value expression, $P \in \Gamma$ is a variable.

$$\text{in}_i \ \frac{\Gamma \vdash P_i}{\Gamma \vdash \Sigma_I P} \ i{:}I \qquad \text{tup} \ \frac{\forall i{:}I,\ \Gamma \vdash P_i}{\Gamma \vdash \bigotimes_I P}$$

$$\text{var} \ \frac{P \in \Gamma}{\Gamma \vdash P} \qquad \text{thunk} \ \frac{\Gamma \vdash N}{\Gamma \vdash \Box N}$$

- A derivation of $\Gamma \vdash N$ is a computation expression.
- Computing with values:

$$\text{case} \ \frac{\Gamma \vdash \Sigma_I P \quad \forall i{:}I,\ \Gamma.P_i \vdash N}{\Gamma \vdash N} \qquad \text{split} \ \frac{\Gamma \vdash \bigotimes_I P \quad \Gamma.\overline{P_i}^{i:I} \vdash N}{\Gamma \vdash N}$$

$$\text{let} \ \frac{\Gamma \vdash P \quad \Gamma.P \vdash N}{\Gamma \vdash N} \qquad \text{force} \ \frac{\Gamma \vdash \Box N}{\Gamma \vdash N}$$

CBPV typing: computations

- Computation introductions:

$$\text{return}\ \frac{\Gamma \vdash P}{\Gamma \vdash \diamond P} \qquad \lambda\ \frac{\Gamma.P \vdash N}{\Gamma \vdash P \Rightarrow N} \qquad \text{record}\ \frac{\forall i{:}I,\ \Gamma \vdash N_i}{\Gamma \vdash \Pi_I N}$$

- Computation eliminations:

$$\text{bind}\ \frac{\Gamma \vdash \diamond P \quad \Gamma.P \vdash N}{\Gamma \vdash N} \qquad \text{app}\ \frac{\Gamma \vdash P \Rightarrow N \quad \Gamma \vdash P}{\Gamma \vdash N}$$

$$\text{proj}_i\ \frac{\Gamma \vdash \Pi_I N}{\Gamma \vdash N_i}$$

CBPV: effect

- Let $1 = \otimes_{\varnothing}\ \_$ be the unit type.
- Simple effect: print string literals.

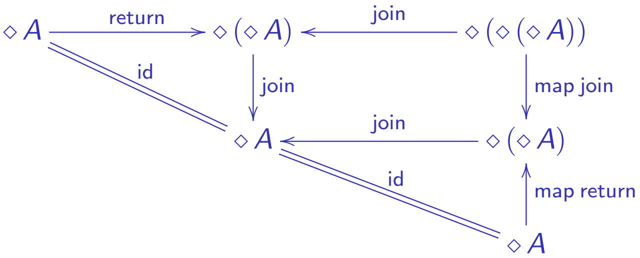
$$\mathsf{print}\, s \; \frac{}{\Gamma \vdash \diamond 1}$$

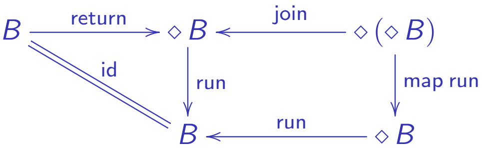
- Example (using bind infix):

$$f = \mathsf{print}\, \texttt{"function"}\, \mathsf{bind}\ \_.\lambda x.\mathsf{print}\, \texttt{"argument"}\, \mathsf{bind}\ \_.\mathsf{return}\, (\mathsf{var}\, x)$$

- CBPV effects are lazy, can only be observed at value types.
- f by itself prints nothing; app f _ prints "function" and "argument".

Monad and monad algebras





CBPV: algebraic semantics

- Assume a cartesian closed category with coproducts.
- Contexts $P_1 \dots P_n$ are products $P_1 \times \dots \times P_n$ and $\Gamma \vdash A$ homsets.
- Interpret $\diamond$ as a strong monad; $\Box$ as identity (for now).
- Interpret each N as monad algebra $\mathsf{run}_N : \diamond N \to N$.

$$
\begin{array}{lcl}
\mathsf{run}_{\diamond P} & : & \diamond(\diamond P) \to \diamond P \\
\mathsf{run}_{\diamond P} & = & \mathsf{join} \\[1ex]
\mathsf{run}_{P \Rightarrow N} & : & \diamond(P \Rightarrow N) \to P \Rightarrow N \\
\mathsf{run}_{P \Rightarrow N}\; m & = & \lambda p.\, \mathsf{run}_N\, (\mathsf{map}\, (\lambda f.\, \mathsf{app}\; f\; p)\; m) \\[1ex]
\mathsf{run}_{\Pi_I N} & : & \diamond(\Pi_I N) \to \Pi_I N \\
\mathsf{run}_{\Pi_I N}\; m & = & \mathsf{record}\; i.\, \mathsf{run}_N\, (\mathsf{map}\; \mathsf{proj}_i\; m)
\end{array}
$$

- This interprets binding in N (going via $\diamond N$):

$$
\mathsf{bind}\ \frac{\Gamma \vdash \diamond P \quad \Gamma.P \vdash N}{\Gamma \vdash N}
$$

Graded effects

- $\Gamma \vdash N \mid e$ where e is drawn from an effect pomonoid.
- Example: keep track of the length of the output.

$$\text{print}\, s \;\frac{}{\Gamma \vdash \diamond\top \mid n}\; n = |s|$$

- Effects accumulate.

$$\text{return} \;\frac{\Gamma \vdash P}{\Gamma \vdash \diamond P \mid 0} \qquad \text{bind} \;\frac{\Gamma \vdash \diamond P \mid e_1 \quad \Gamma.P \vdash N \mid e_2}{\Gamma \vdash N \mid e_1 + e_2}$$

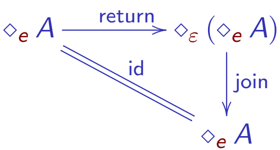
- Effects get frozen and subsumed.

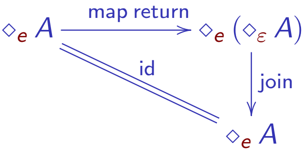
$$\text{thunk} \;\frac{\Gamma \vdash N \mid e}{\Gamma \vdash [e]N} \qquad \text{force} \;\frac{\Gamma \vdash [e]N}{\Gamma \vdash N \mid e} \qquad \text{sub} \;\frac{\Gamma \vdash N \mid e}{\Gamma \vdash N \mid e'}\; e \leqslant e'$$

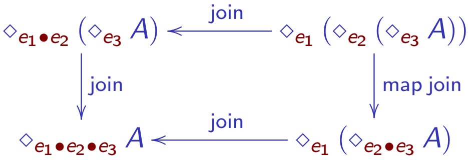
Graded monads

$$
\begin{array}{lcl}
\text{return} & : & A \to \Diamond_{\varepsilon} A \\
\text{join} & : & \Diamond_{e_1}(\Diamond_{e_2} A) \to \Diamond_{e_1 \bullet e_2} A \\
\Diamond_{e \leqslant e'} & : & \Diamond_{e} A \to \Diamond_{e'} A
\end{array}
$$

$$
\begin{array}{rcl}
\Diamond_n A & = & \mathrm{Char}^n \times A \\
a & \mapsto & ("", a) \\
(s_1, (s_2, a)) & \mapsto & (s_1 + s_2, a)
\end{array}
$$







Graded monad algebras

$$
\begin{array}{lcl}
B_{e \leqslant e'} & : & B_e \to B_{e'} \\
\mathsf{run}_B & : & \Diamond_{e_1} B_{e_2} \to B_{e_1 \bullet e_2}
\end{array}
$$

$$
\begin{array}{ccc}
B_e & \xrightarrow{\ \mathsf{return}\ } & \Diamond_\varepsilon B_e \\
 & \overset{\mathsf{id}}{\searrow\!\!\searrow} & \downarrow \mathsf{run} \\
 & & B_e
\end{array}
\qquad
\begin{array}{ccc}
\Diamond_{e_1 \bullet e_2} B_{e_3} & \xleftarrow{\ \mathsf{join}\ } & \Diamond_{e_1} (\Diamond_{e_2} B_{e_3}) \\
\downarrow \mathsf{run} & & \downarrow \mathsf{map\ run} \\
B_{e_1 \bullet e_2 \bullet e_3} & \xleftarrow{\ \mathsf{run}\ } & \Diamond_{e_1} B_{e_2 \bullet e_3}
\end{array}
$$

$$
\begin{array}{lcl}
(\Diamond P)_e & = & \Diamond_e P \\
(P \Rightarrow N)_e & = & P \Rightarrow N_e \\
(\Pi_{i:I} N_i)_e & = & \Pi_{i:I} (N_i)_e
\end{array}
$$

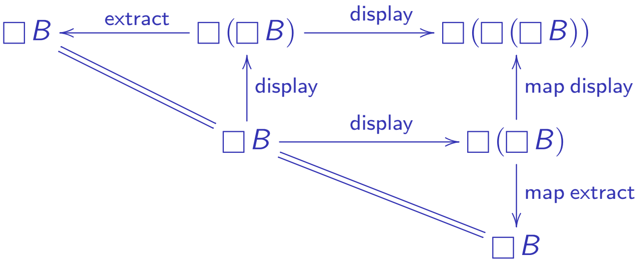
Graded effect semantics

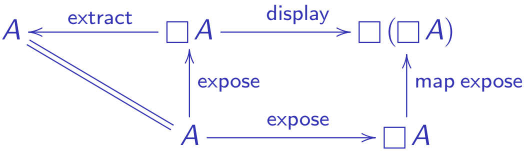
- Negative types N are interpreted as functors from effects to objects.
- Positive types P are interpreted as objects; $[e]N = N_e$.
- $\Gamma \vdash P$ is $\Gamma \to P$.
- $\Gamma \vdash N \mid e$ is $\Gamma \to N_e$.

$$\text{return}\ \frac{\Gamma \to P}{\Gamma \to \diamond_\varepsilon P} \qquad \text{bind}\ \frac{\Gamma \to \diamond_{e_1} P \quad \Gamma \times P \to N_{e_2}}{\Gamma \to N_{e_1 \bullet e_2}}$$

bind justified by $\Gamma \to \Gamma \times \diamond_{e_1} P \to \diamond_{e_1}(\Gamma \times P) \to \diamond_{e_1} N_{e_2} \to N_{e_1 \bullet e_2}$.

Comonads and comonad coalgebras





Comonadic semantics

- Judgements $P_1 \ldots P_n \vdash A$ are homsets $P_1 \times \ldots \times P_n \to A$.
- $\Box$ is interpreted as monoidal comonad.
- Positive types P are $\Box$-coalgebras with $P \to \Box P$.

$$
\begin{array}{lcl}
\mathrm{expose}_{\Box N} & : & \Box N \to \Box(\Box N) \\
\mathrm{expose}_{\Box N} & = & \mathrm{display} \\
\mathrm{expose}_{\Sigma_I P} & : & \Sigma_I P \to \Box(\Sigma_I P) \\
\mathrm{expose}_{\Sigma_I P}(i, v) & = & \mathrm{map}\ \mathrm{in}_i\ (\mathrm{expose}_{P_i}\, v) \\
\mathrm{expose}_{\otimes_I P} & : & \otimes_I P \to \Box(\otimes_I P) \\
\mathrm{expose}_{\otimes_I P}(v_i)_{i:I} & = & \mathrm{zip}\ (\mathrm{expose}_{P_i}\, v_i)_{i:I} \\
\mathrm{zip} & : & \otimes_{i:I}(\Box B_i) \to \Box(\otimes_I B) \\
\mathrm{expose}_{\Gamma} & : & \Gamma \to \Box\Gamma
\end{array}
$$

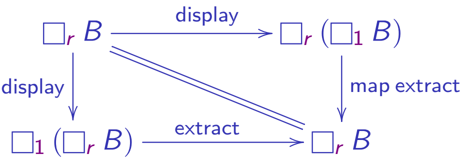
- Thunking needs monoidality.

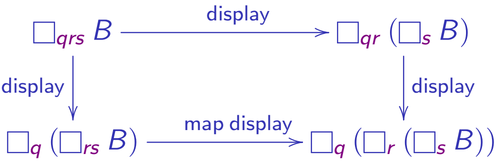
$$\text{thunk}\ \frac{\Gamma \to N}{\Gamma \to \Box N} \qquad \text{force}\ \frac{\Gamma \to \Box N}{\Gamma \to N}$$

$$\begin{array}{lcl} \text{thunk} & : & \Gamma \to \Box\Gamma \to \Box N \\ \text{force} & : & \Gamma \to \Box N \to N \end{array}$$

Graded comonad

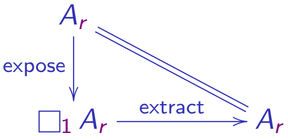
$$\begin{array}{lcl} \text{extract} & : & \Box_1 B \to B \\ \text{display} & : & \Box_{qr} B \to \Box_q (\Box_r B) \end{array}$$

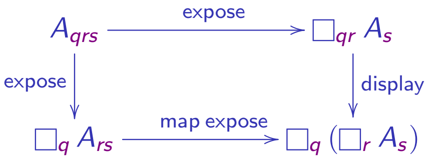




Graded comonad coalgebras

$$
\begin{array}{lcl}
A_{q \leqslant r} & : & A_q \to A_r \\
\text{expose} & : & A_{qr} \to \Box_q A_r
\end{array}
$$





Semiring-graded comonads

- Switch to a closed symmetric monoidal category.
- Additive commutative monoid on grades maps to monoidal structure.

$$\begin{array}{lcl} \mathsf{drop} & : & \Box_0 B \to \mathsf{I} \\ \mathsf{duplicate} & : & \Box_{r+s} B \to \Box_r B \otimes \Box_s B \end{array}$$

- Extends to coalgebras.

$$\begin{array}{lcl} \mathsf{drop}_A & : & A_0 \to \mathsf{I} \\ \mathsf{duplicate}_A & : & A_{r+s} \to A_r \otimes A_s \end{array}$$

Semantics of positive types

Positive types P are interpreted as graded $\Box$-coalgebras.

$$
\begin{array}{lcl}
(\Box N)_r & = & \Box_r N \\
(\Sigma_I P)_r & = & \Sigma_{i:I} (P_i)_r \\
(\otimes_I P)_r & = & \otimes_{i:I} (P_i)_r
\end{array}
$$

$$
\begin{array}{lcl}
\mathsf{expose}_{\Box N} & : & (\Box N)_{qr} \to \Box_q (\Box N)_r \\
\mathsf{expose} & = & \mathsf{display} \\
\mathsf{expose}_{\Sigma_I P} & : & (\Sigma_I P)_{qr} \to \Box_q (\Sigma_I P)_r \\
\mathsf{expose}_{\Sigma_I P}(i, v) & = & \mathsf{map}\ \mathsf{in}_i\ (\mathsf{expose}_{P_i} v) \\
\mathsf{expose}_{\otimes_I P} & : & (\otimes_I P)_{qr} \to \Box_q (\otimes_I P)_r \\
\mathsf{expose}_{\otimes_I P}(v_i)_{i:I} & = & \mathsf{zip}\ (\mathsf{expose}_{P_i} v_i)_{i:I}
\end{array}
$$

Graded contexts

- Contexts Γ come with grading contexts γ of the same length.
- $r_1 P_1 \ldots r_n P_n = (r_1 \ldots r_n)(P_1 \ldots P_n)$.
- Grading contexts γ ("vectors") form a *left module* to grades.
- Write e_x for the unit vector $\mathsf{e}_x(y) = 1$ iff $x = y$ else 0.
- Contexts Γ are interpreted as *structured* graded $\Box$-coalgebras Γ_γ.

$$
\begin{array}{lcccc}
\mathsf{expose}_\Gamma & : & \Gamma_{r\gamma} & \to & \Box_r \Gamma_\gamma \\
\mathsf{drop}_\Gamma & : & \Gamma_0 & \to & \mathsf{I} \\
\mathsf{duplicate}_\Gamma & : & \Gamma_{\gamma+\delta} & \to & \Gamma_\gamma \otimes \Gamma_\delta \\
\mathsf{duplicate}_\Gamma & : & \Gamma_{\sum_I \gamma} & \to & \bigotimes_{i:I} \Gamma_{\gamma_i}
\end{array}
$$

Coeffect-graded CBPV

- Negative types $N ::= \langle r \rangle P \mid r P \multimap N \mid \Pi_{i:I} N_i$.
- $\gamma\Gamma \vdash P$ and $\gamma\Gamma \vdash N$.

$$\text{var}\ \frac{x : P \in \Gamma}{e_x\Gamma \vdash P} \qquad \text{let}\ \frac{\delta\Gamma \vdash P \quad \gamma\Gamma.rP \vdash N}{(\gamma + r\delta)\Gamma \vdash N}$$

$$\text{in}_i\ \frac{\gamma\Gamma \vdash P_i}{\gamma\Gamma \vdash \Sigma_I P}\ i{:}I \qquad \text{case}\ \frac{\delta\Gamma \vdash \Sigma_I P \quad \forall i{:}I,\ \gamma\Gamma.rP_i \vdash N}{(\gamma + r\delta)\Gamma \vdash N}$$

$$\text{tup}\ \frac{\forall i{:}I,\ \gamma_i\Gamma \vdash P_i}{(\sum_I \gamma)\Gamma \vdash \bigotimes_I P} \qquad \text{split}\ \frac{\delta\Gamma \vdash \bigotimes_I P \quad \gamma\Gamma.\overline{rP_i}^{i:I} \vdash N}{(\gamma + r\delta)\Gamma \vdash N}$$

$$\text{thunk}\ \frac{\gamma\Gamma \vdash N}{\gamma\Gamma \vdash \Box N} \qquad \text{force}\ \frac{\gamma\Gamma \vdash \Box N}{\gamma\Gamma \vdash N}$$

CBPV typing: computations

$$\text{return}\ \frac{\gamma\Gamma \vdash P}{r\gamma\Gamma \vdash \langle r\rangle P} \qquad \text{bind}\ \frac{\delta\Gamma \vdash \langle r\rangle P \quad \gamma\Gamma.rP \vdash N}{(\gamma+\delta)\Gamma \vdash N}$$

$$\lambda\ \frac{\gamma\Gamma.rP \vdash N}{\gamma\Gamma \vdash rP \multimap N} \qquad \text{app}\ \frac{\gamma\Gamma \vdash rP \multimap N \quad \delta\Gamma \vdash P}{(\gamma + r\delta)\Gamma \vdash N}$$

$$\text{record}\ \frac{\forall i{:}I,\ \gamma\Gamma \vdash N_i}{\gamma\Gamma \vdash \Pi_I N} \qquad \text{proj}_i\ \frac{\gamma\Gamma \vdash \Pi_I N}{\gamma\Gamma \vdash N_i}$$

Semantics of coeffect-grading

- Computations $\gamma\Gamma \vdash N$ are morphisms $\Gamma_\gamma \to N$.
- Values $\gamma\Gamma \vdash P$ are morphisms families $\Gamma_{r\gamma} \to P_r$ natural in r.

$$\text{var}\ \frac{x : P \in \Gamma}{\mathsf{e}_x\Gamma \vdash P} \qquad \text{let}\ \frac{\delta\Gamma \vdash P \quad \gamma\Gamma.rP \vdash N}{(\gamma + r\delta)\Gamma \vdash N}$$

$$\begin{array}{lcl} \text{var} & : & (\Gamma.P.\Delta)_{r(\vec{0}.1.\vec{0})} \to \Gamma_{\vec{0}} \otimes P_r \otimes \Delta_{\vec{0}} \to P_r \\ \text{let} & : & \Gamma_{\gamma+r\delta} \to \Gamma_\gamma \otimes \Gamma_{r\delta} \to \Gamma_\gamma \otimes P_r \to (\Gamma.P)_{\gamma.r} \to N \end{array}$$

$$\text{thunk}\ \frac{\gamma\Gamma \vdash N}{\gamma\Gamma \vdash \Box N} \qquad \text{force}\ \frac{\gamma\Gamma \vdash \Box N}{\gamma\Gamma \vdash N}$$

$$\begin{array}{lcl} \text{thunk} & : & \Gamma_{r\gamma} \to \Box_r \Gamma_\gamma \to \Box_r N = (\Box N)_r \\ \text{force} & : & \Gamma_\gamma = \Gamma_{1\gamma} \to (\Box N)_1 = \Box_1 N \to N \end{array}$$

Semantics ctd.

- Semantics of monadic computations.

$$\text{return}\ \frac{\gamma\Gamma \vdash P}{r\gamma\Gamma \vdash \langle r\rangle P} \qquad \text{bind}\ \frac{\delta\Gamma \vdash \langle r\rangle P \quad \gamma\Gamma.rP \vdash N}{(\gamma + \delta)\Gamma \vdash N}$$

$$\begin{array}{lcl} \text{return} & : & \Gamma_{r\gamma} \to P_r \to \diamond P_r = \langle r\rangle P \\ \text{bind} & : & \Gamma_{\gamma+\delta} \to \Gamma_\gamma \otimes \Gamma_\delta \to \Gamma_\gamma \otimes (\diamond P_r) \to \diamond(\Gamma_\gamma \otimes P_r) \to \diamond N \to N \end{array}$$

- Semantics of functions.

$$\lambda\ \frac{\gamma\Gamma.rP \vdash N}{\gamma\Gamma \vdash rP \multimap N} \qquad \text{app}\ \frac{\gamma\Gamma \vdash rP \multimap N \quad \delta\Gamma \vdash P}{(\gamma + r\delta)\Gamma \vdash N}$$

$$\text{app} \quad : \quad \Gamma_{\gamma+r\delta} \to \Gamma_\gamma \otimes \Gamma_{r\delta} \to \Gamma_\gamma \otimes P_r \to N$$

Fully graded CBPV

- Judgements $\gamma\Gamma \vdash P$ and $\gamma\Gamma \vdash N \mid e$.
- Interpreted as $\Gamma_{r\gamma} \to P_r$ and $\Gamma_\gamma \to N_e$.
- Monad $(\langle r \rangle P)_e = \Diamond_e P_r$ and comonad $([e]N)_r = \Box_r N_e$.