Graded monads and quantified computational effects

Tarmo Uustalu
Reykjavik University and Tallinn University of Technology

FMCS 2018, Sackville, NS, 31 May-2 June 2018

What is this?

- We organize effectful computations with monads, “idioms” (lax monoidal endofunctors), arrows, relative monads etc.
- Often it is useful to track the “degree” of effectfulness, e.g., for ensuring safety (honoring of given resource usage bounds) or optimizations.
- Enter *grading* of monads, idioms etc.
- This is revisiting the old idea of effect systems and in particular of the marriage of monads and effects (with effect inference and all that).
- But this time we are guided by a mathematical foundation.

Married monads and effects: Graded monadic metalanguage

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash \mathrm{ret}\, t : T\, \mathrm{i}\, A}$$

$$\frac{\Gamma \vdash t : T\, e\, A \quad \Gamma, x : A \vdash u : T\, f\, B}{\Gamma \vdash \mathrm{let\ ret}\ x \leftarrow t\ \mathrm{in}\ u : T\,(e * f)\, B}$$

$$\left(\frac{\Gamma \vdash t : T\, e\, B \quad e \leq e'}{\Gamma \vdash t : T\, e'\, B} \right)$$

$$\frac{\Gamma \vdash t : T\, \mathrm{Bool} \quad \Gamma \vdash u : T\, f\, A \quad \Gamma \vdash v : T\, f'\, A}{\Gamma \vdash \mathrm{if}\ t\ \mathrm{then}\ u\ \mathrm{then}\ u' : T\,(e * (f \vee f'))\, A}$$

$$\frac{}{\Gamma \vdash \mathrm{vanish} : T\, \mathrm{o}\, A} \qquad \frac{\Gamma \vdash u : T\, e\, A \quad \Gamma \vdash v : T\, e'\, A}{\Gamma \vdash u\ \mathrm{or}\ u' : T\,(e + e')\, A}$$

Outline

- Graded monads
  - Kleisli and Eilenberg-Moore categories for graded monads
- Graded "monads of monoids" (MonadPlus instances)
- Graded distributive laws of monads over monads

Graded monads

- Given a *monoid* $(E, \mathrm{i}, *)$. A *graded monad* on a category $\mathbb{C}$ is
  - for any $e : E$, a functor $T\,e : \mathbb{C} \to \mathbb{C}$
  - a nat. transf. $\eta : \mathrm{Id} \to T\,\mathrm{i}$
  - for any $e, f : E$, a nat. transf. $\mu^{e,f} : T\,e \cdot T\,f \to T\,(e * f)$

  such that

$$
\begin{tikzcd}
\mathrm{Id} \cdot T\,e \arrow[r, "\eta \cdot T\,e"] \arrow[d, equal] & T\,\mathrm{i} \cdot T\,e \arrow[d, "\mu^{\mathrm{i},e}"] \\
T\,e \arrow[r, equal] & T\,(\mathrm{i} * e)
\end{tikzcd}
\qquad
\begin{tikzcd}
T\,e \cdot \mathrm{Id} \arrow[r, "T\,e \cdot \eta"] \arrow[d, equal] & T\,e \cdot T\,\mathrm{i} \arrow[d, "\mu^{e,\mathrm{i}}"] \\
T\,e \arrow[r, equal] & T\,(e * \mathrm{i})
\end{tikzcd}
$$

$$
\begin{tikzcd}
T\,e \cdot (T\,f \cdot T\,g) \arrow[r, equal] \arrow[d, "T\,e \cdot \mu^{f,g}"'] & (T\,e \cdot T\,f) \cdot T\,g \arrow[r, "\mu^{e,f} \cdot T\,g"] & T\,(e * f) \cdot T\,g \arrow[d, "\mu^{e*f,g}"] \\
T\,e \cdot T\,(f * g) \arrow[r, "\mu^{e,f*g}"] & T\,(e * (f * g)) \arrow[r, equal] & T\,((e * f) * g)
\end{tikzcd}
$$

- In short, a graded monad on $\mathbb{C}$ is a lax monoidal functor from $(E, \mathrm{i}, *)$ as a discrete monoidal category to $([\mathbb{C}, \mathbb{C}], \mathrm{Id}, \cdot)$.

Graded monads ctd

- It is useful to generalize from a monoid to a *pomonoid* $((E, \leq), \mathrm{i}, *)$, i.e., a set E with a partial order $\leq$ and a monoid structure $(\mathrm{i}, *)$ such that $*$ is monotone wrt. $\leq$.
- NB! i need not be the least element (nor the greatest).
- Then a *graded monad* has also
  - for any $e \leq e'$, a natural transformation $T\,(e \leq e') : T\,e \to T\,e'$

  such that

$$T\,(e \leq e) = \mathrm{id}_{T\,e}$$
$$T\,(e' \leq e'') \circ T\,(e \leq e') = T\,(e \leq e' \leq e'')$$

$$\begin{tikzcd}
T\,e \cdot T\,f \arrow[r, "\mu^{e,f}"] \arrow[d, "T\,(e \leq e') \cdot T\,(f \leq f')"'] & T\,(e * f) \arrow[d, "T\,(e * f \leq e' * f')"] \\
T\,e' \cdot T\,f' \arrow[r, "\mu^{e',f'}"] & T\,(e' * f')
\end{tikzcd}$$

- Again, a graded monad is a lax monoidal functor, this time from a thin monoidal category.
- One can also grade with a general monoidal category.

Example: Graded maybe

$E = \{\mathrm{pure}, \mathrm{fail}, \mathrm{mf}\}$

$$\begin{tikzcd}
& \mathrm{mf} \arrow[dl, no head] \arrow[dr, no head] & \\
\mathrm{pure} & & \mathrm{fail}
\end{tikzcd}$$

$\mathsf{i} = \mathrm{pure}$

$*$	pure	fail	mf
pure	pure	fail	mf
fail	fail	fail	fail
mf	mf	fail	mf

$$\begin{aligned}
T\,\mathrm{pure} \quad X &= X \\
T\,\mathrm{fail} \quad X &= 1 \\
T\,\mathrm{mf} \quad X &= X + 1
\end{aligned}$$

$$T\,(\mathrm{pure} \leq \mathrm{mf}) \quad X = \quad X \xrightarrow{\mathsf{inl}} X + 1$$

$$T\,(\mathrm{fail} \leq \mathrm{mf}) \quad X = \quad 1 \xrightarrow{\mathsf{inr}} X + 1$$

Example: Graded state

Given a set S of states.

$$E = \{\mathrm{pure}, \mathrm{ro}, \mathrm{wo}^+, \mathrm{wo}, \mathrm{rw}\}$$

$$\begin{tikzcd}
& \mathrm{rw} \arrow[dl, dash] \arrow[dr, dash] & & \\
\mathrm{ro} \arrow[dr, dash] & & \mathrm{wo} \arrow[dl, dash] \arrow[dr, dash] & \\
& \mathrm{pure} & & \mathrm{wo}^+
\end{tikzcd}$$

$\mathsf{i} = \mathrm{pure}$

$*$	pure	ro	wo^+	wo	rw
pure	pure	ro	wo^+	wo	rw
ro	ro	ro	rw	rw	rw
wo^+	wo^+	wo^+	wo^+	wo^+	wo^+
wo	wo	rw	wo^+	wo	rw
rw	rw	rw	rw	rw	rw

$$\begin{array}{lll}
T\,\mathrm{pure} & X = & X \\
T\,\mathrm{ro} & X = & S \Rightarrow X \\
T\,\mathrm{wo}^+ & X = & S \times X \\
T\,\mathrm{wo} & X = & (S+1) \times X \cong (S \times X) + X \\
T\,\mathrm{rw} & X = & S \Rightarrow S \times X
\end{array}$$

Example: Graded writer

Given an alphabet Σ.

Option 1:

$(E, \leq, \mathrm{i}, *) = (\mathbb{N}, \leq, 0, +)$

$T\,n\,X = \Sigma^{\leq n} \times X$

Option 2:

$(E, \leq, \mathrm{i}, *) = (\mathcal{P}\Sigma^*, \subseteq, [], +\!\!+)$
(or we could use any class of languages containing Σ^* and closed under $[]$ and $+\!\!+$, eg regular languages)

where
$[] = \{[]\}$
$L +\!\!+ L' = \{w +\!\!+ w' \mid w \in L, w' \in L'\}$

$T\,L\,X = L \times X$

Kleisli category of a graded monad

- Given a pomonoid $(\mathbb{E}, \mathrm{i}, *)$ and a graded monad T on $\mathbb{C}$.
- An object of the *Kleisli category* of T is given by
  - an element e of E,
  - an object X of $\mathbb{C}$.
- A map between (e, X), (e', Y) is given by
  - an element f of E such that $e * f \leq e'$
  - a map $k : X \to T\, f\, Y$

  modulo the equivalence relation $\sim$ given by the rule

$$\frac{f \leq f'}{(f, e * f \leq e * f' \leq e', k) \sim (f', e * f' \leq e', T\,(f \leq f') \circ k)}$$

Eilenberg-Moore category of a graded monad

- An object of the *E-M category* of T (an algebra) is given by
  - for any $e : E$, an object $A\,e$
  - for any $e \leq e'$, a map $A\,(e \leq e') : A\,e \to A\,e'$,
  - for any $e, f : E$, a map $a^{e,f} : T\,e\,(A\,f) \to A\,(e * f)$

  such that

$$A\,(e \leq e) = \mathrm{id}_{A\,e} \qquad A\,(e' \leq e'') \circ A\,(e \leq e') = A\,(e \leq e' \leq e'')$$

$$\begin{tikzcd}
T\,e\,(A\,f) \arrow[r, "a^{e,f}"] \arrow[d, "T\,(e\leq e')\,(A\,(f\leq f'))"'] & A\,(e*f) \arrow[d, "A\,(e*f\leq e'*f')"] \\
T\,e'\,(A\,f') \arrow[r, "a^{e',f'}"] & A\,(e'*f')
\end{tikzcd}
\quad
\begin{tikzcd}
A\,e \arrow[r, "\eta_{A\,e}"] \arrow[dr, equal] & T\,\mathrm{i}\,(A\,e) \arrow[d, "a^{\mathrm{i},e}"] \\
 & A\,e
\end{tikzcd}
\quad
\begin{tikzcd}
T\,e\,(T\,f\,(A\,g)) \arrow[r, "\mu^{e,f}_{A\,g}"] \arrow[d, "T\,e\,a^{f,g}"'] & T\,(e*f)\,(A\,g) \arrow[d, "a^{e*f,g}"] \\
T\,e\,(A\,(f*g)) \arrow[r, "a^{e,f*g}"] & A\,(e*f*g)
\end{tikzcd}$$

- A morphism (algebra map) between (A, a), (B, b) is given by
  - for any $e : E$, a map $h^e : A\,e \to B\,e$

  such that

$$\begin{tikzcd}
A\,e \arrow[r, "h^e"] \arrow[d, "A\,(e\leq e')"'] & B\,e \arrow[d, "B\,(e\leq e')"] \\
A\,e' \arrow[r, "h^{e'}"] & B\,e'
\end{tikzcd}
\qquad
\begin{tikzcd}
T\,e\,(A\,f) \arrow[r, "a^{e,f}"] \arrow[d, "T\,e\,h^f"'] & A\,(e*f) \arrow[d, "h^{e*f}"] \\
T\,e\,(B\,f) \arrow[r, "b^{e,f}"] & B\,(e*f)
\end{tikzcd}$$

Resolutions of graded monads

- A *resolution* of T is given by
  - a category $\mathbb{D}$,
  - a strict monoidal functor $A : (\mathbb{E}, \mathsf{i}, *) \to ([\mathbb{D}, \mathbb{D}], \mathsf{Id}, \cdot)$,
  - adjoint functors L, R between $\mathbb{C}$ and $\mathbb{D}$

  such that
  - $T\, e = R \cdot A\, e \cdot L$
  - (appropriate conditions on η, μ)
- The Kleisli category is the initial resolution, the E-M category is the final resolution.

Graded monads of monoids (MonadPlus instances)

- Given a *right near-semiring* $(E, \mathsf{i}, *, \mathsf{o}, +)$,
  i.e., a set E with two monoid structures $(\mathsf{i}, *)$, $(\mathsf{o}, +)$,
  with $*$ distributing over o and $+$ from the right.
  (Left distributivity and commutativity of $+$ are not required.)
- A *graded monad of monoids* on a category $(\mathbb{C}, 1, \times)$ with finite products is an $(E, \mathsf{i}, *)$-graded monad on $\mathbb{C}$ with
  - a nat. transf. $\mathsf{e} : 1 \to T\,\mathsf{o}$
  - a nat. transf. $\mathsf{m}^{e,f} : T\,e \times T\,f \to T\,(e+f)$

  such that

$$
\begin{tikzcd}
1 \times T\,e \arrow[r, "\mathsf{e} \times T\,e"] \arrow[d, "\lambda_{T\,e}"'] & T\,\mathsf{o} \times T\,e \arrow[d, "\mathsf{m}^{\mathsf{o},e}"] \\
T\,e \arrow[r, equal] & T\,(\mathsf{o} + e)
\end{tikzcd}
\qquad
\begin{tikzcd}
T\,e \times 1 \arrow[r, "T\,e \times \mathsf{e}"] & T\,e \times T\,\mathsf{o} \arrow[d, "\mathsf{m}^{e,\mathsf{o}}"] \\
T\,e \arrow[u, "\rho_{T\,e}"] \arrow[r, equal] & T\,(e + \mathsf{o})
\end{tikzcd}
$$

$$
\begin{tikzcd}
T\,e \times (T\,f \times T\,g) \arrow[d, "T\,e \times \mathsf{m}^{f,g}"'] & (T\,e \times T\,f) \times T\,g \arrow[l, "\alpha_{T\,e, T\,f, T\,g}"'] \arrow[r, "\mathsf{m}^{e,f} \times T\,g"] & T\,(e+f) \times T\,g \arrow[d, "\mathsf{m}^{e+f,g}"] \\
T\,e \times T\,(f+g) \arrow[r, "\mathsf{m}^{e,f+g}"] & T\,(e + (f+g)) \arrow[r, equal] & T\,((e+f)+g)
\end{tikzcd}
$$

Graded monads of monoids ctd

$$\begin{tikzcd}
1 \arrow[r, equal] \arrow[d, equal] & 1 \cdot T\,g \arrow[r, "\mathrm{e} \cdot T\,g"] & T\,\mathrm{o} \cdot T\,g \arrow[d, "\mu^{\mathrm{o},g}"] \\
1 \arrow[rr, "\mathrm{e}"] & & T\,\mathrm{o} \arrow[r, equal] & T\,(\mathrm{o} * g)
\end{tikzcd}$$

$$\begin{tikzcd}
T\,e \cdot T\,g \times T\,f \cdot T\,g \arrow[r, equal] \arrow[d, "\mu^{e,g} \times \mu^{f,g}"'] & (T\,e \times T\,f) \cdot T\,g \arrow[r, "\mathrm{m}^{e,f} \cdot T\,g"] & T\,(e+f) \cdot T\,g \arrow[d, "\mu^{e+f,g}"] \\
T\,(e * g) \times T(f * g) \arrow[r, "\mathrm{m}^{e*g,f*g}"] & T\,(e * g + f * g) \arrow[r, equal] & T\,((e+f) * g)
\end{tikzcd}$$

- In short, a graded monad of monoids is a lax right near-semiring functor from $(E, \mathrm{i}, *, \mathrm{o}, +)$ as a discrete semiring category to $([\mathbb{C}, \mathbb{C}], \mathrm{Id}, \cdot, 1, \times)$.

- The finite product structure $(1, \times)$ on $\mathbb{C}$ (which is lifted to $[\mathbb{C}, \mathbb{C}]$) can be replaced with a general monoidal structure $(I, \otimes)$.

- Similarly to the monad case, it makes sense to generalize to grading with an ordered right near-semiring or with a general right near-semiring category.

Example: Graded nondeterminism

$(\mathbb{E}, \mathsf{i}, *, \mathsf{o}, +) = ((\mathbb{N}, \leq), 1, *, 0, +)$

$T\, n\, X = X^{\leq n}$

$(\mathbb{E}, \mathsf{i}, *, \mathsf{o}, +) = ((\mathbb{N}, \geq), 1, *, 0, +)$

$T\, n\, X = X^{\geq n}$

Composing graded monads: Matching pairs of actions

- Given two monoids $(E_0, \mathrm{i}_0, *_0)$ and $(E_1, \mathrm{i}_1, *_1)$.
- A *matching pair* is a pair of functions $\searrow\colon E_1 \times E_0 \to E_0$, $\swarrow\colon E_1 \times E_0 \to E_1$ such that

$$\begin{gathered}
e_1 \searrow \mathrm{i}_0 = \mathrm{i}_0 \\
e_1 \searrow (e_0 *_0 e_0') = (e_1 \searrow e_0) *_0 ((e_1 \swarrow e_0) \searrow e_0') \\
\mathrm{i}_1 \searrow e_0 = e_0 \\
(e_1 *_1 e_1') \searrow e_0 = e_1 \searrow (e_1' \searrow e_0)
\end{gathered}$$

$$\begin{gathered}
e_1 \swarrow \mathrm{i}_0 = e_1 \\
e_1 \swarrow (e_0 *_0 e_0') = (e_1 \swarrow e_0) \swarrow e_0' \\
\mathrm{i}_1 \swarrow e_0 = \mathrm{i}_1 \\
(e_1 *_1 e_1') \swarrow e_0 = (e_1 \swarrow (e_1' \searrow e_0)) *_1 (e_1' \swarrow e_0)
\end{gathered}$$

- A matching pair equips $E_0 \times E_1$ with a monoid structure by $\mathrm{i} = (\mathrm{i}_0, \mathrm{i}_1)$ and $(e_0, e_1) * (e_0', e_1') = (e_0 *_0 (e_1 \searrow e_0'), (e_1 \swarrow e_0') *_1 e_1')$, a *Zappa-Szép product* structure on $E_0 \times E_1$.
- Matching pairs and Zappa-Szép product structures are in a bijection.

Graded distributive laws

- Given two monoids $(E_0, \mathrm{i}_0, *_0)$, $(E_1, \mathrm{i}_1, *_1)$ with a matched pair and graded monads (T_0, η_0, μ_0) and (T_1, η_1, μ_1).
- A *graded distributive law* consists of, for any $e_1 : E_1$, $e_0 : E_0$, a nat. transf. $\theta^{e_1,e_0} : T_1\, e_1 \cdot T_0\, e_0 \to T_0\,(e_1 \searrow e_0) \cdot T_1\,(e_1 \swarrow e_0)$ such that

$$\begin{tikzcd}
T_1\, e_1 \arrow[rr, equal] \arrow[d, "T_1\, e_1 \cdot \eta_0"'] & & T_1\, e_1 \arrow[d, "\eta_0 \cdot T_1\, e_1"] \\
T_1\, e_1 \cdot T_0\, \mathrm{i}_0 \arrow[r, "\theta^{e_1, \mathrm{i}_0}"] & T_0\,(e_1 \searrow \mathrm{i}_0) \cdot T_1\,(e_1 \swarrow \mathrm{i}_0) \arrow[r, equal] & T_0\, \mathrm{i}_0 \cdot T_1\, e_1
\end{tikzcd}$$

and three more equations hold

- Let $T\,(e_0, e_1) = T_0\, e_0 \cdot T_1\, e_1$. A graded distributive law equips T with a graded monad structure for the Zappa-Szép product by

$$\eta = \eta_0 \cdot \eta_1$$

$$\mu^{(e_0,e_1),(e_0',e_1')} = \mu_0^{e_0, e_1 \searrow e_0'} \cdot \mu_1^{e_1 \swarrow e_0', e_1'} \circ T_0\, e_0 \cdot \theta^{e_1, e_0'} \cdot T_1\, e_1$$

  a *compatible graded monad* structure.
- Distributive laws and compatible graded monad structures are in a bijection.

Example: Distributing graded maybe

Let $(\mathbb{E}_1, \mathrm{i}_1, *_1)$ and (T_1, η_1, μ_1) be the pomonoid and graded monad from the graded maybe example.

For any pomonoid $(\mathbb{E}_0, \mathrm{i}_0, *_0)$ that has joins and graded monad (T_0, η_0, μ_0), the following is a matching pair for which we have a graded distributive law:

$$
\begin{array}{rcl}
\mathrm{pure} \searrow e_0 & = & e_0 \\
\mathrm{fail} \searrow e_0 & = & \mathrm{i}_0 \\
\mathrm{mf} \searrow e_0 & = & e_0 \vee \mathrm{i}_0
\end{array}
\qquad
\begin{array}{rcl}
e_1 \swarrow e_0 & = & e_1
\end{array}
$$

$$\theta^{e_1, e_0} : T_1\, e_1 \cdot T_0\, e_0 \to T_0\, (e_1 \searrow e_0) \cdot T_1\, (e_1 \swarrow e_0)$$

$$\theta_X^{\mathrm{pure}, e_0} : \; T_0\, e_0\, X = T_0\, e_0\, X$$

$$\theta_X^{\mathrm{fail}, e_0} : \; 1 \xrightarrow{\eta_{01}} T_0\, i_0\, 1$$

$$\theta_X^{\mathrm{mf}, e_0} : \; T_0\, e_0\, X + 1 \xrightarrow{T_0\, e_0\, X + \eta_{01}} T_0\, e_0\, X + T_0\, i_0\, 1 \longrightarrow T_0\, (e_0 \vee i_0)\, (X + 1)$$

Grading the stack writer monad

$$(\mathbb{E}_0, \mathsf{i}_0, *_0) = ((\mathbb{N}, \geq), 0, +)$$
$$(\mathbb{E}_1, \mathsf{i}_1, *_1) = ((\mathbb{N}, \leq), 0, +)$$
$$n_1 \searrow n_0 = n_0 \mathbin{\dot{-}} n_1$$
$$n_1 \swarrow n_0 = n_1 \mathbin{\dot{-}} n_0$$

$$\mathsf{i} = (0, 0)$$
$$(n_0, n_1) * (n_0', n_1') = (n_0 + (n_0' \mathbin{\dot{-}} n_1), (n_1 \mathbin{\dot{-}} n_0') + n_1)$$

$$T_0\, n_0\, X = \mathbb{N}_{\geq n_0} \times X$$
$$T_1\, n_1\, X = \Sigma^{\leq n_1} \times X$$

$$\begin{array}{cccccc} \theta_X^{n_0,n_1} & : & \Sigma^{\leq n_1} \times (\mathbb{N}_{\geq n_0} \times X) & \to & \mathbb{N}_{\geq n_0 \dot{-} n_1} \times (\Sigma^{\leq n_1 \dot{-} n_0} \times X) \\ & & (w, (k, x)) & \mapsto & (k \mathbin{\dot{-}} |w|, (\mathrm{drop}\; k\; w, xs)) \end{array}$$

Takeaway

- Graded monads etc are natural concepts both theoretically and in terms of programming examples.
- Marrying monads and effects works!
- But as ever we see that it pays off to look at the categorical generalities to get things right.